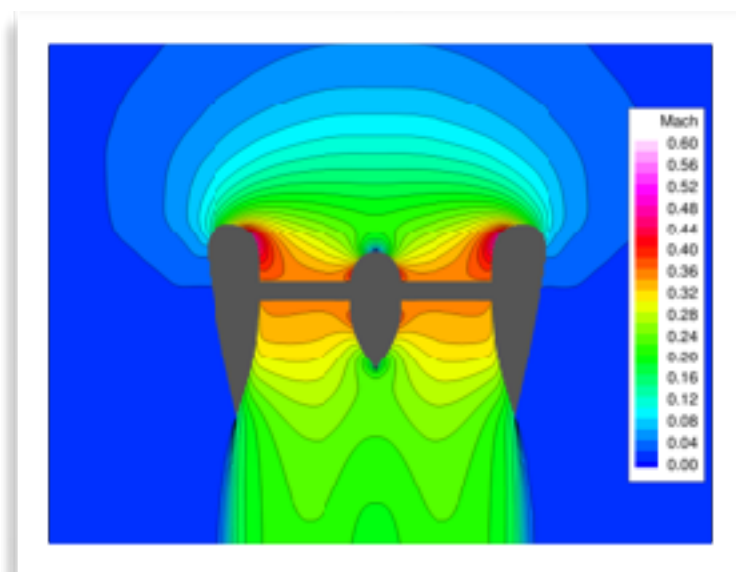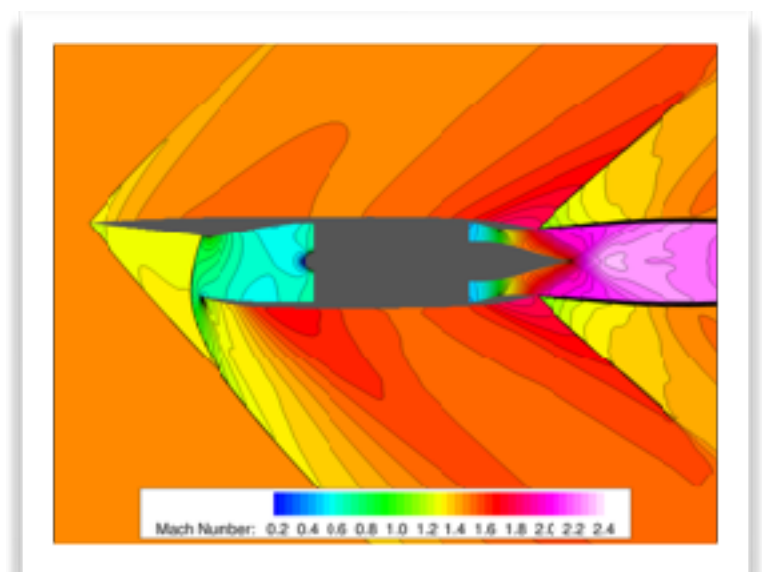# Formulation and Implementation of Inflow/Outflow Boundary Conditions to Simulate Propulsive Effects

David L Rodriguez
Michael Aftosmis
Marian Nemec
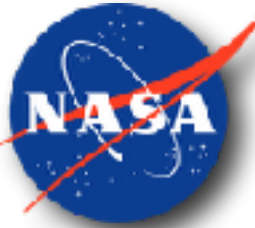
January 8, 2018
SciTech 2018

NASA Ames Research Center

Computational Aerosciences Branch   *Advanced Supercomputing Division*

Cart3D Aerodynamic Analysis & Design Package

- Automated multilevel Cartesian mesh generation with adjoint-driven adaptive refinement

- Cut-cell approach in cells that include model surface

- Finite volume, 2nd-order accurate Euler solver with explicit Runge-Kutta time stepping and multigrid

- Steady or time-accurate

- Part of a design framework allowing for gradient-based aerodynamic shape optimization of user-specified functional

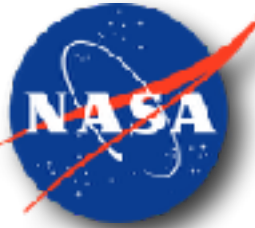- Shown to be highly effective for analysis and design of low boom aircraft

$h = 80ft$
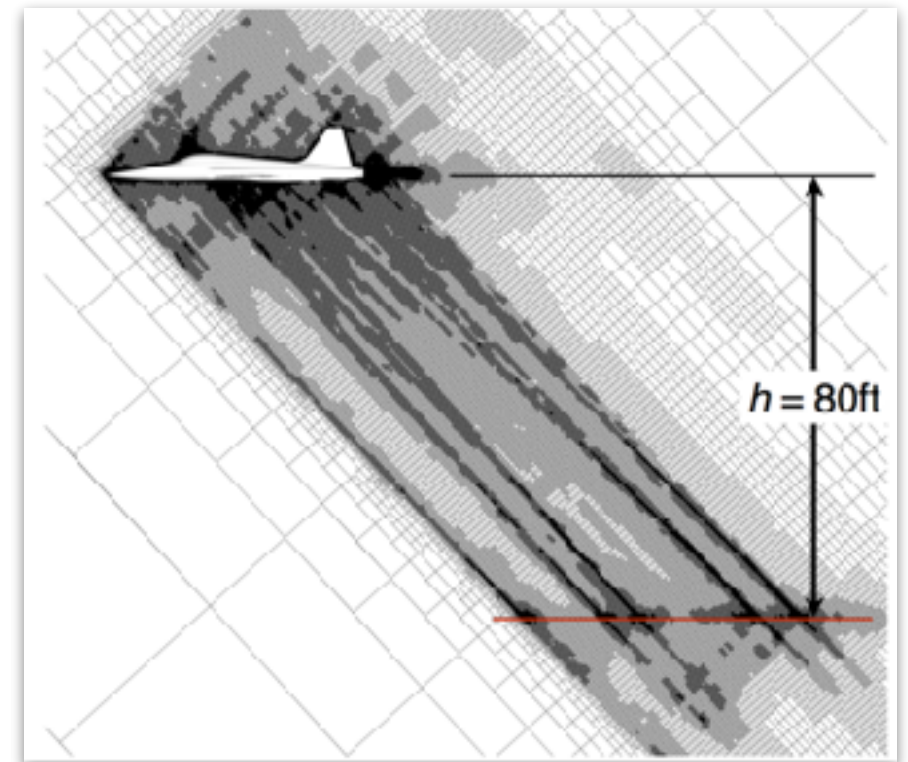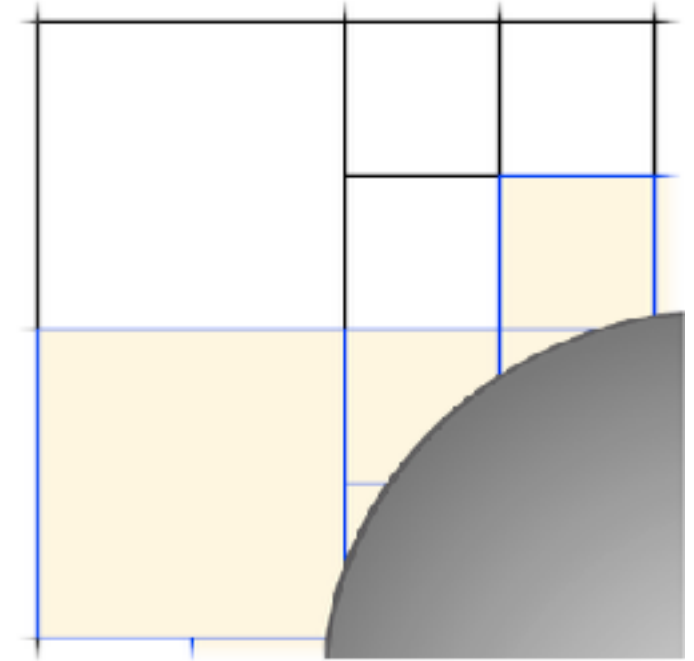
Motivation for New Boundary Conditions

- Current **SurfBC** inflow/outflow boundary condition requires user to specify an entire state (ρ, u, v, w, p) at the boundary

  - Riemann solver is applied to compute flux at the boundary and thus boundary condition is always well-posed

  - Robust and flexible since it can be used for both inflow and outflow, subsonic and supersonic

- Inconvenient when user wants to specify inflow or outflow with minimal information

  - for subsonic flow through inlets, most common boundary condition is back pressure

  - for subsonic flow into nozzles, most common boundary condition is specifying total pressure and total temperature (and flow direction)

  - very difficult to specify mass flow rate, particularly in cases where nonlinear flow features are prevalent
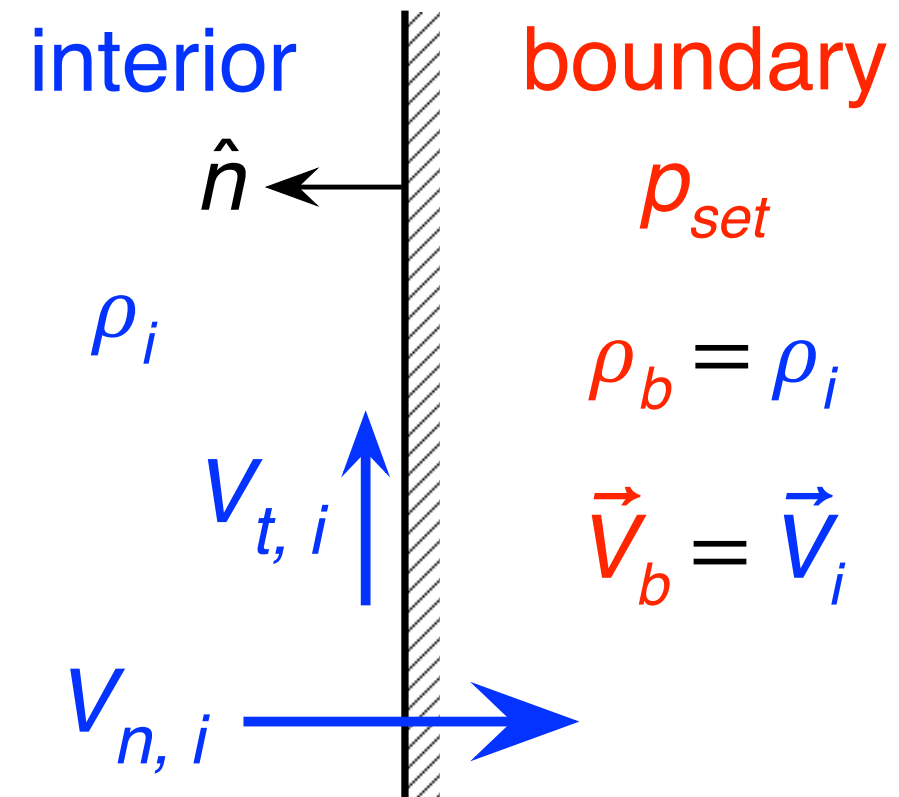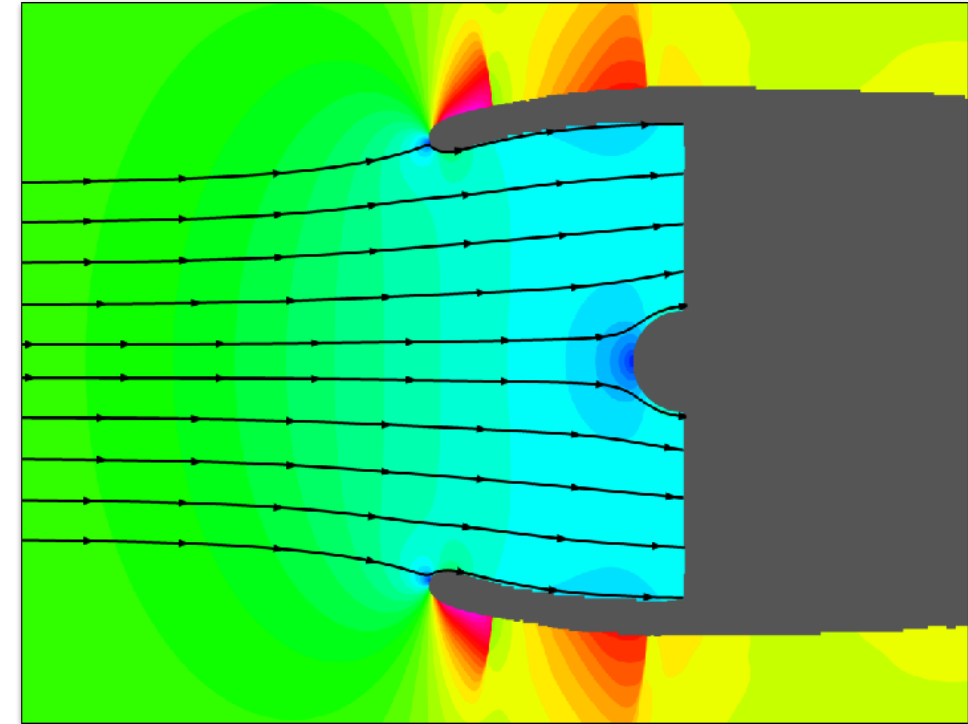
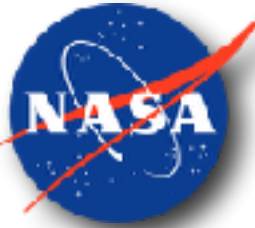# Back Pressure Outflow BC Formulations
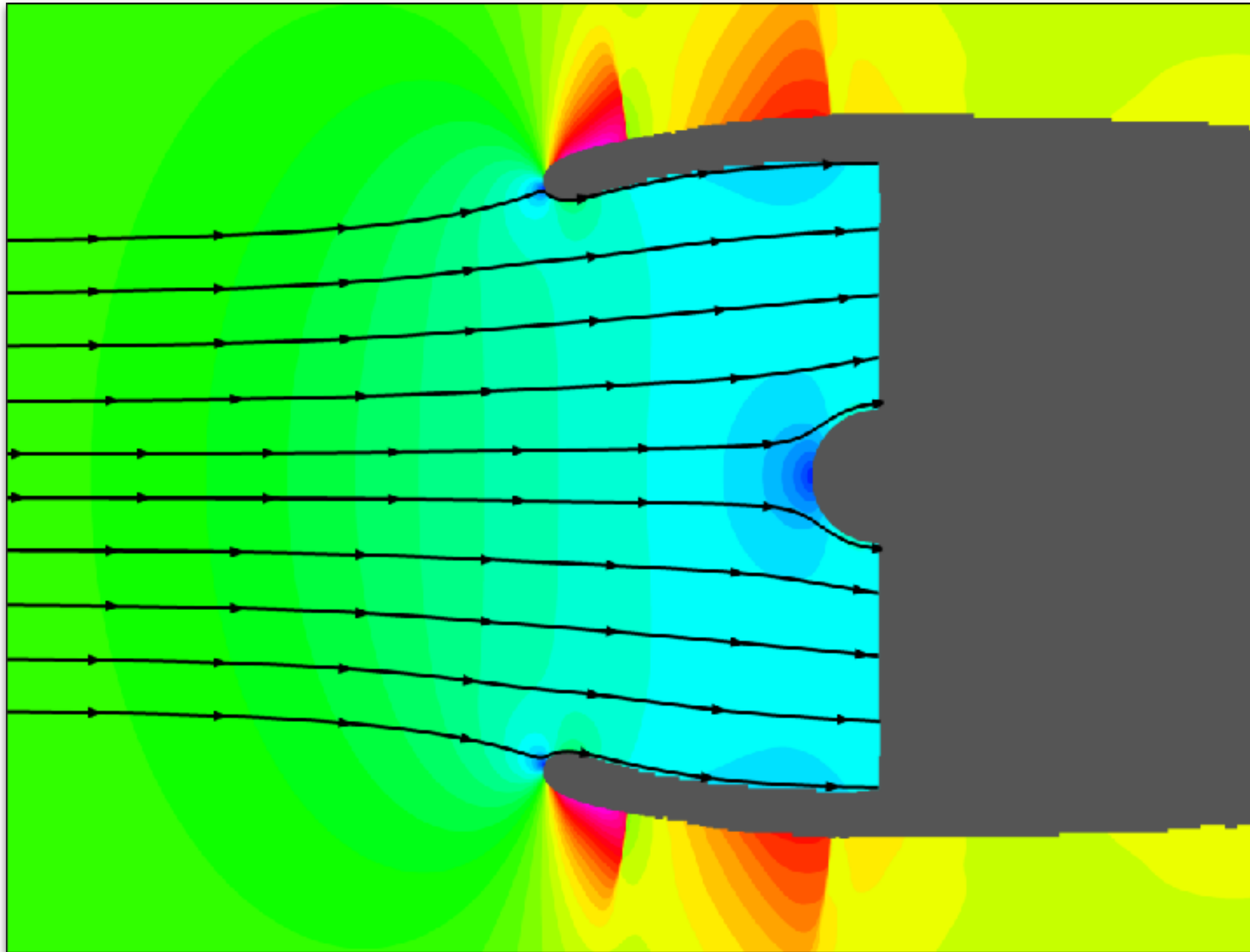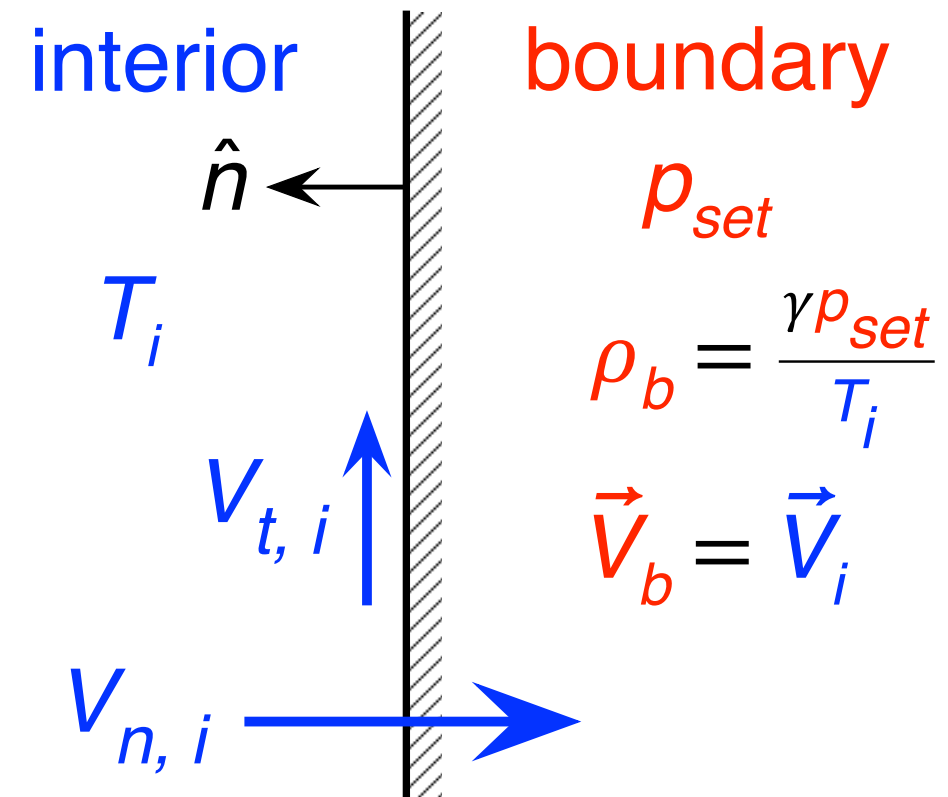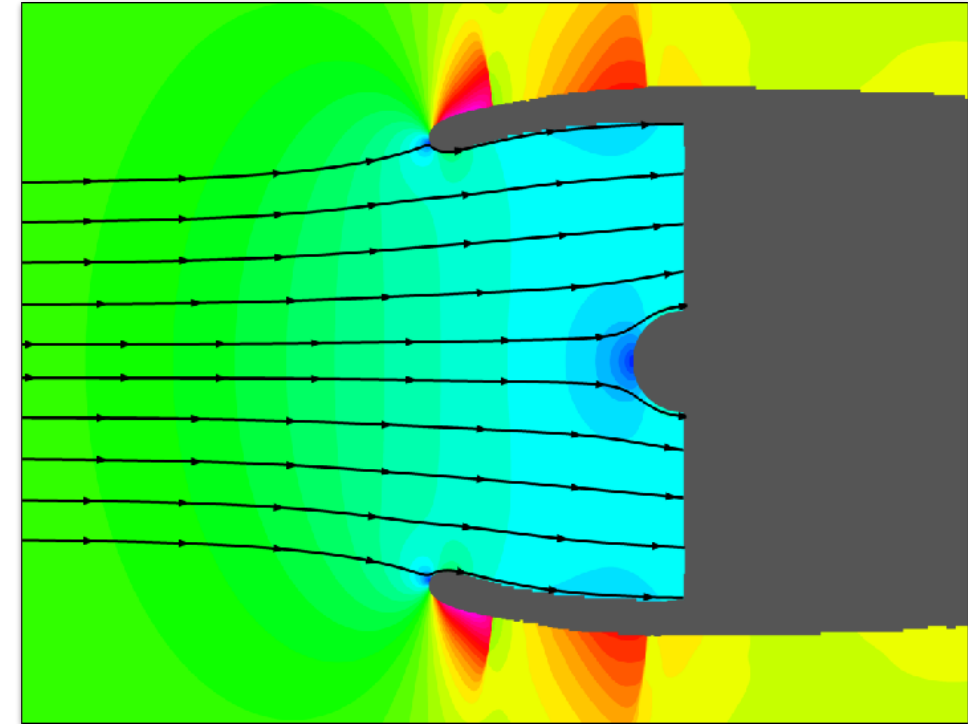
Back Pressure Outflow BC Formulations

- CFL3D

  - set pressure

  - extrapolate density and velocity

interior		boundary
$\hat{n} \leftarrow$		p_{set}
ρ_i		$\rho_b = \rho_i$
$V_{t,i}$		$\vec{V}_b = \vec{V}_i$
$V_{n,i}$		

Back Pressure Outflow BC Formulations

- CFL3D
  - set pressure
  - extrapolate density and velocity

- Fun3D
  - set pressure
  - extrapolate velocity and temperature
  - update density

interior | boundary

$$\hat{n} \leftarrow$$

$$p_{set}$$

$$T_i$$

$$\rho_b = \frac{\gamma p_{set}}{T_i}$$

$$v_{t, i}$$

$$\vec{V}_b = \vec{V}_i$$
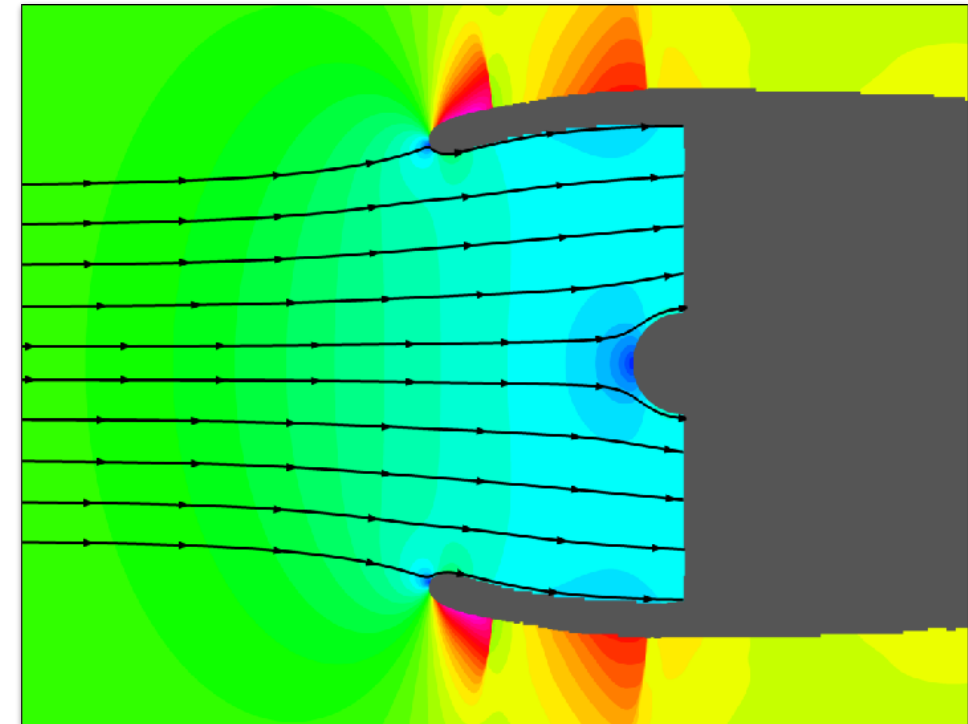
$$V_{n, i}$$

Back Pressure Outflow BC Formulations
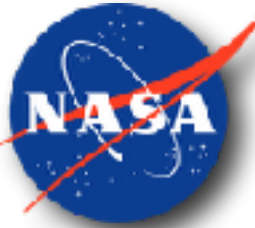
- CFL3D
  - set pressure
  - extrapolate density and velocity
- Fun3D
  - set pressure
  - extrapolate velocity and temperature
  - update density
- OVERFLOW
  - set pressure
  - extrapolate density and momentum
  - update total energy

interior | boundary

$\hat{n}$

p_{set}

ρ_i

$\rho_b = \rho_i$

$(\rho V_t)_i$

$(\rho \vec{V})_b = (\rho \vec{V})_i$

$(\rho e_0)_b = \dfrac{p_{set}}{\gamma\text{-}1} + \dfrac{\|\vec{v}_b\|^2}{2}$

$(\rho V_n)_i$

Back Pressure Outflow BC Formulations

- CFL3D
  - set pressure
  - extrapolate density and velocity
- Fun3D
  - set pressure
  - extrapolate velocity and temperature
  - update density
- OVERFLOW
  - set pressure
  - extrapolate density and momentum
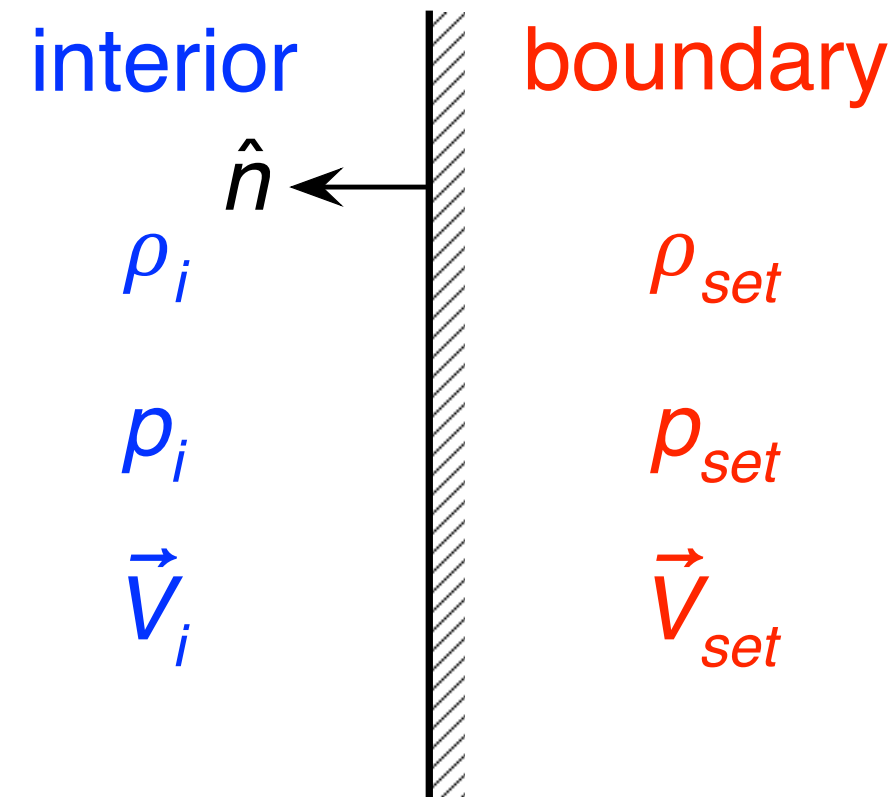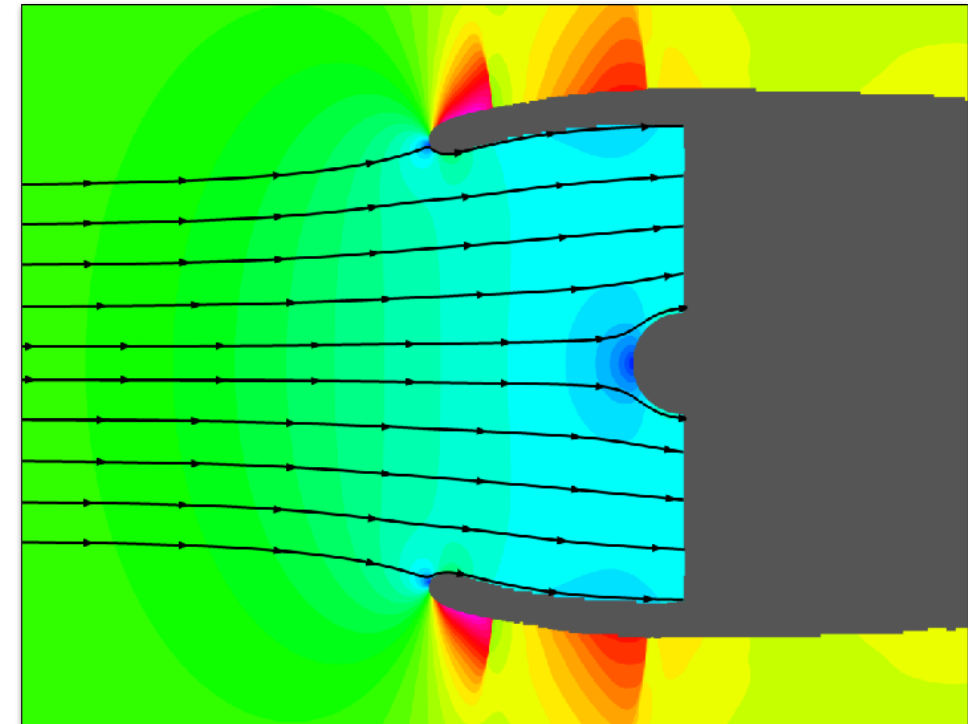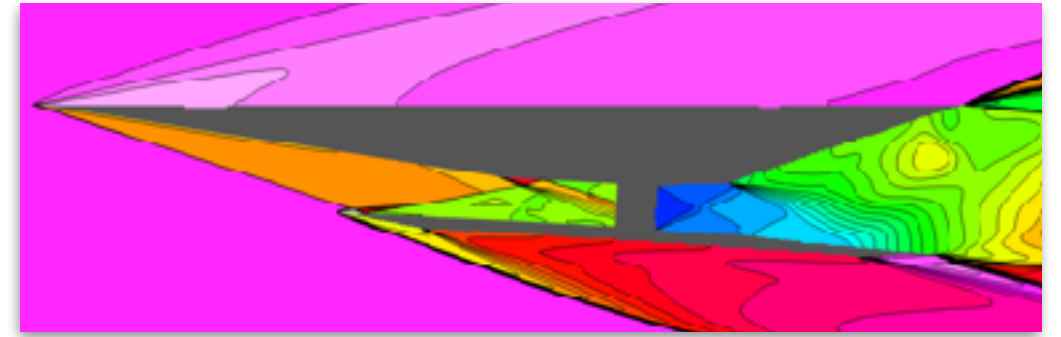 - update total energy
- Cart3D (since 2004)
  - set entire boundary flow state
  - Riemann solver applied

interior | boundary

$\hat{n}$

ρ_i | ρ_{set}

p_i | p_{set}
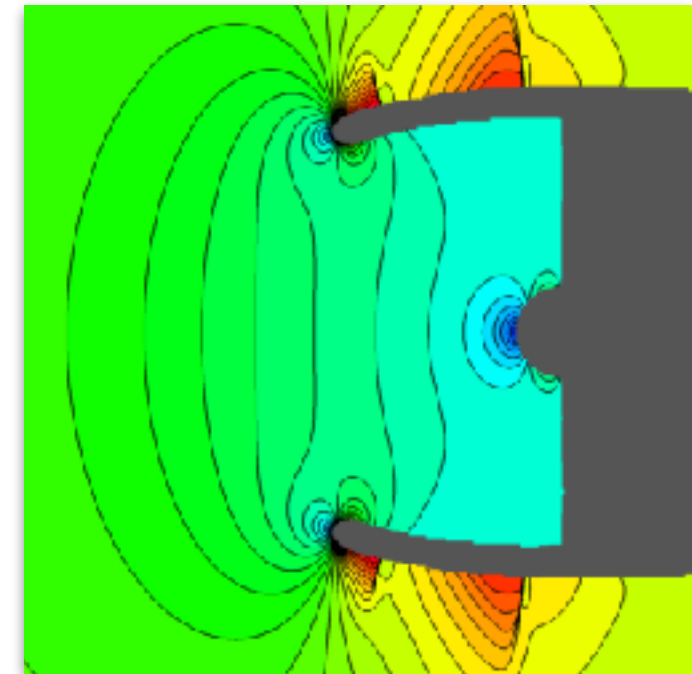
$\vec{V}_i$ | $\vec{V}_{set}$

Cart3D Surface Boundary Conditions

- Solid wall

- Specify full flow state and use Riemann solver (**SurfBC**)

  - Pandya, Murman, Aftosmis, 2004

  - for all inflows and outflows
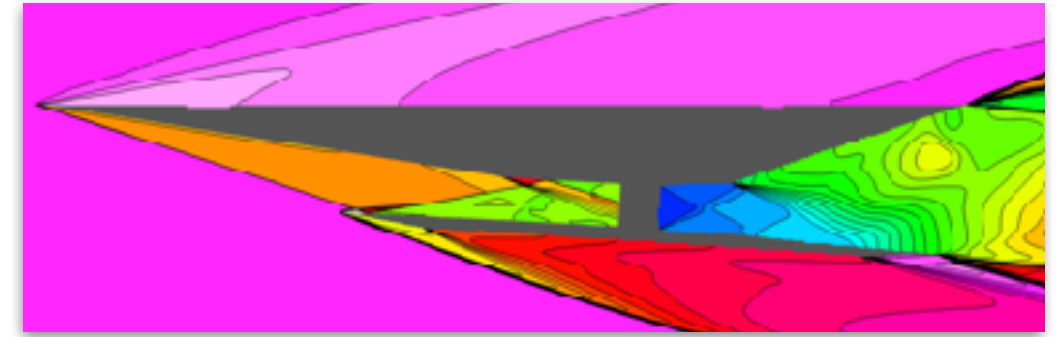
Cart3D Surface Boundary Conditions

- Solid wall

- Specify full flow state and use Riemann solver (**SurfBC**)

  - Pandya, Murman, Aftosmis, 2004

  - for all inflows and outflows

- Subsonic Outflow

  - back pressure **NEW**
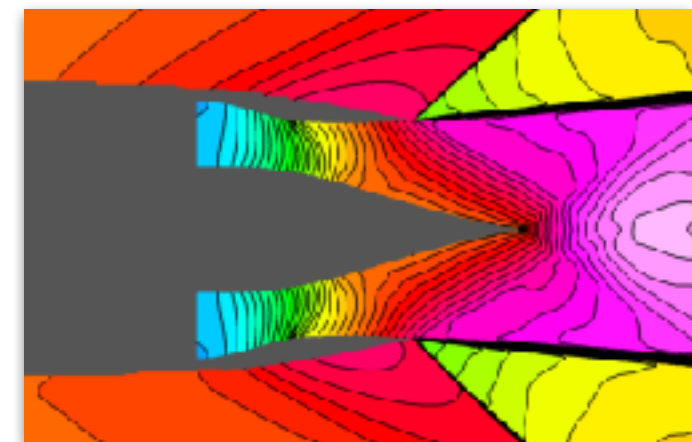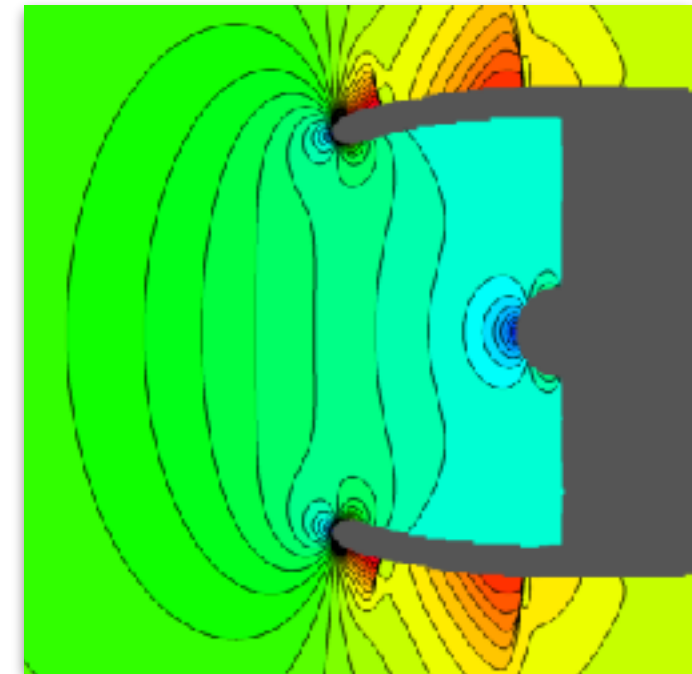
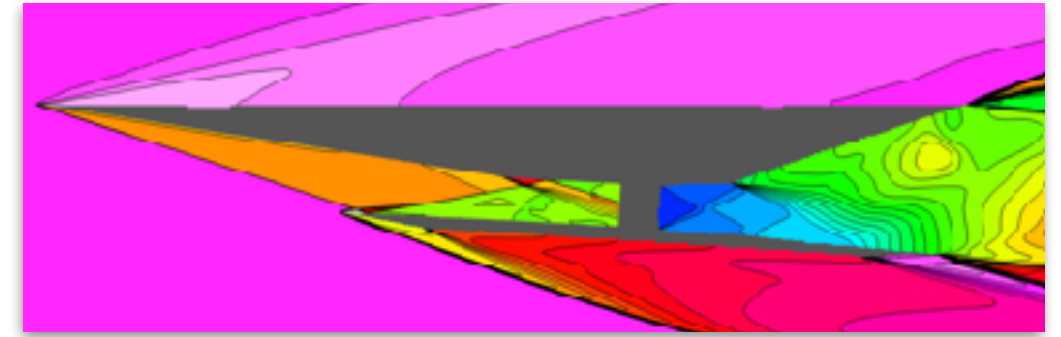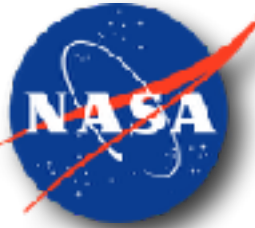 - constant normal velocity **NEW**

Cart3D Surface Boundary Conditions

- Solid wall

- Specify full flow state and use Riemann solver (**SurfBC**)

  - Pandya, Murman, Aftosmis, 2004

  - for all inflows and outflows

- Subsonic Outflow

  - back pressure **NEW**

  - constant normal velocity **NEW**

- Subsonic Inflow

  - total pressure and total temperature **NEW**

  - mass flow rate and total temperature **NEW**
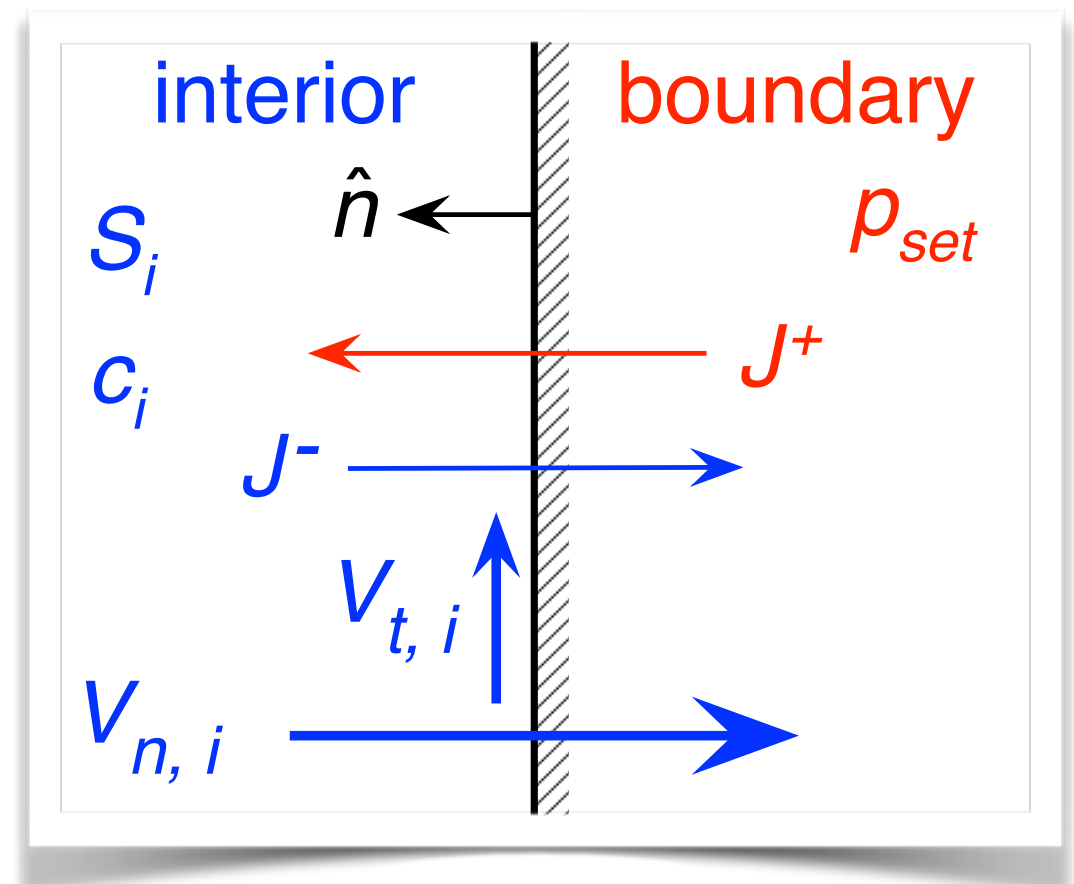
Subsonic Outflow Boundary Conditions

- One flow quantity specified at boundary

  - back pressure

  - normal velocity

- Four flow quantities extrapolated from interior
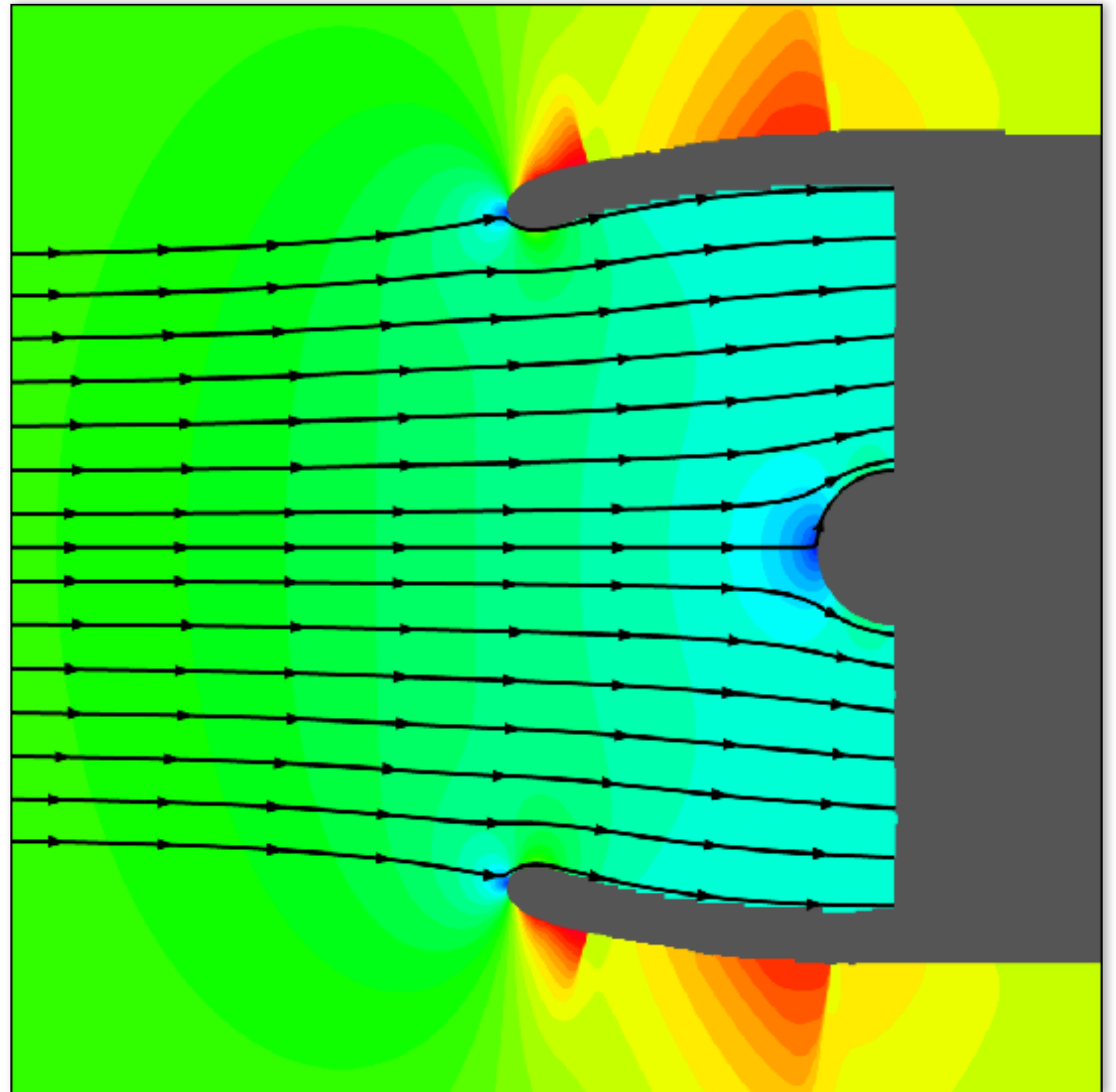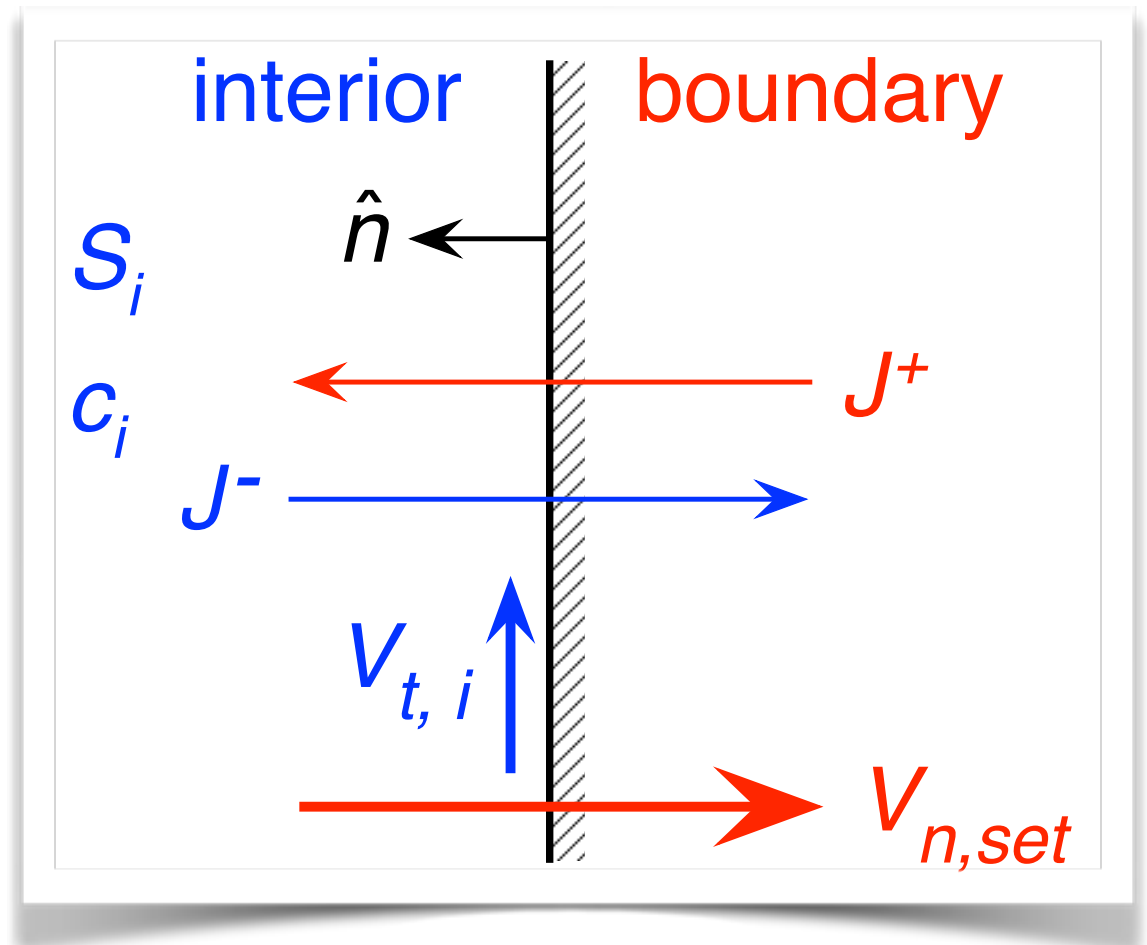
Back Pressure Outflow

- Most other CFD solvers have this common option

- Pressure set to specified value at boundary

- Entropy and tangential velocity extrapolated from interior

- Riemann invariants used to compute boundary state

- Safeguards

  - if flow reverses back into interior (back pressure too high), solid wall boundary enforced

  - if interior flow goes supersonic, compare back pressure to pressure after normal shock occurring at boundary

    - if set back pressure is higher, use after-shock state at boundary, forcing subsonic flow in the interior

    - if set back pressure is lower, extrapolate all flow attributes from interior (supersonic outflow)

- Can be difficult to obtain specific mass flow rate for nonlinear flows

Constant Normal Velocity Outflow

- Allows for robust mass flow rate steering

- Might better represent flow in front of an engine fan face (Pearson '59, Reid '69)

- Normal velocity set to specified value at boundary

- Entropy and tangential velocity extrapolated from interior

- Riemann invariants used to compute boundary state

- Safeguards

  - when interior flow is subsonic but boundary flow is supersonic (bad input velocity), flow is forced to be sonic (choked flow)

  - when interior and boundary flow are both supersonic, supersonic outflow is enforced (all interior quantities extrapolated)

Subsonic Inflow Boundary Conditions

- Four flow quantities specified at boundary

  - velocity set to be normal to boundary (two flow quantities)

  - total pressure and total temperature

  - mass flow rate and total temperature

- One flow quantity extrapolated from interior
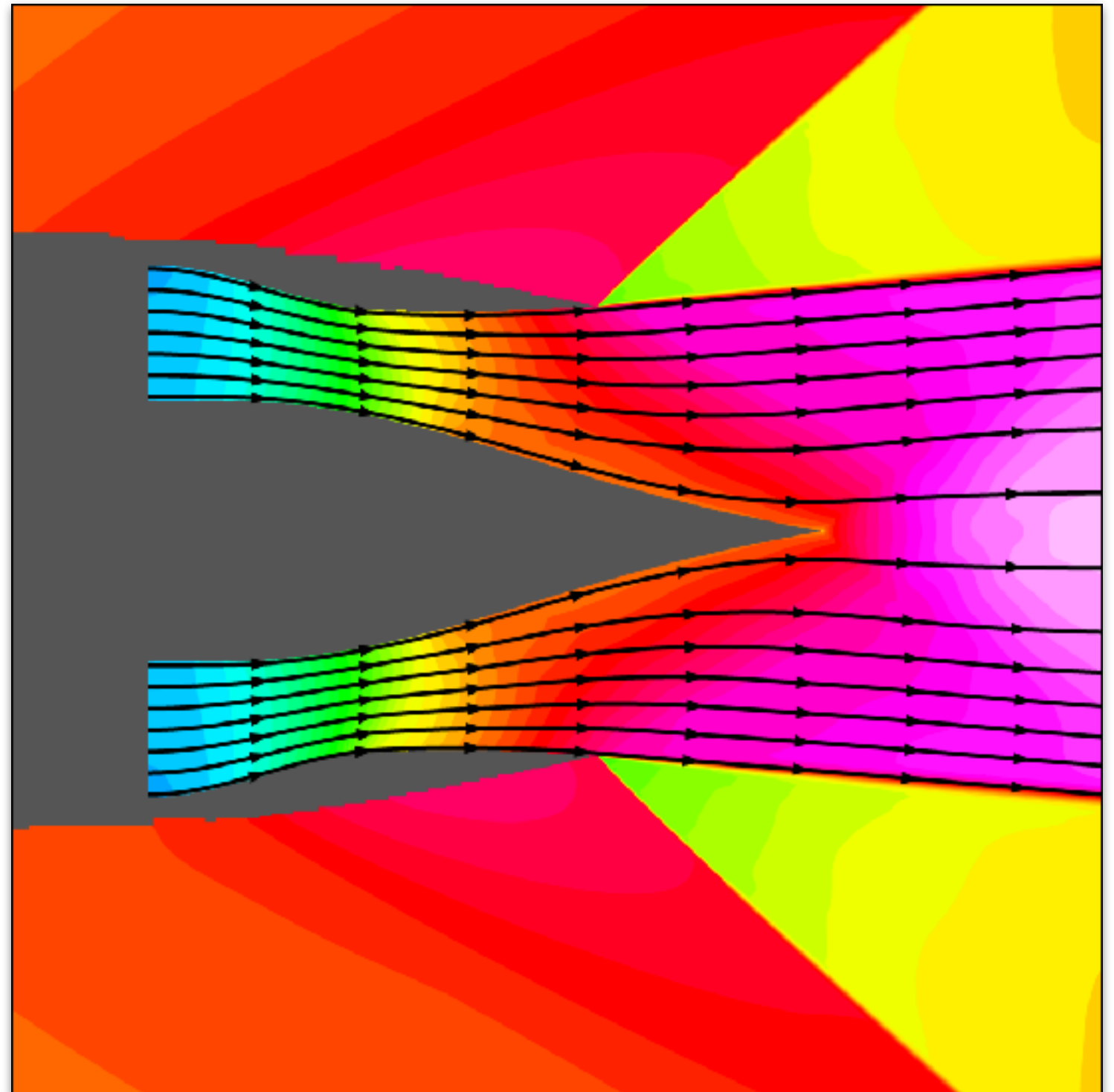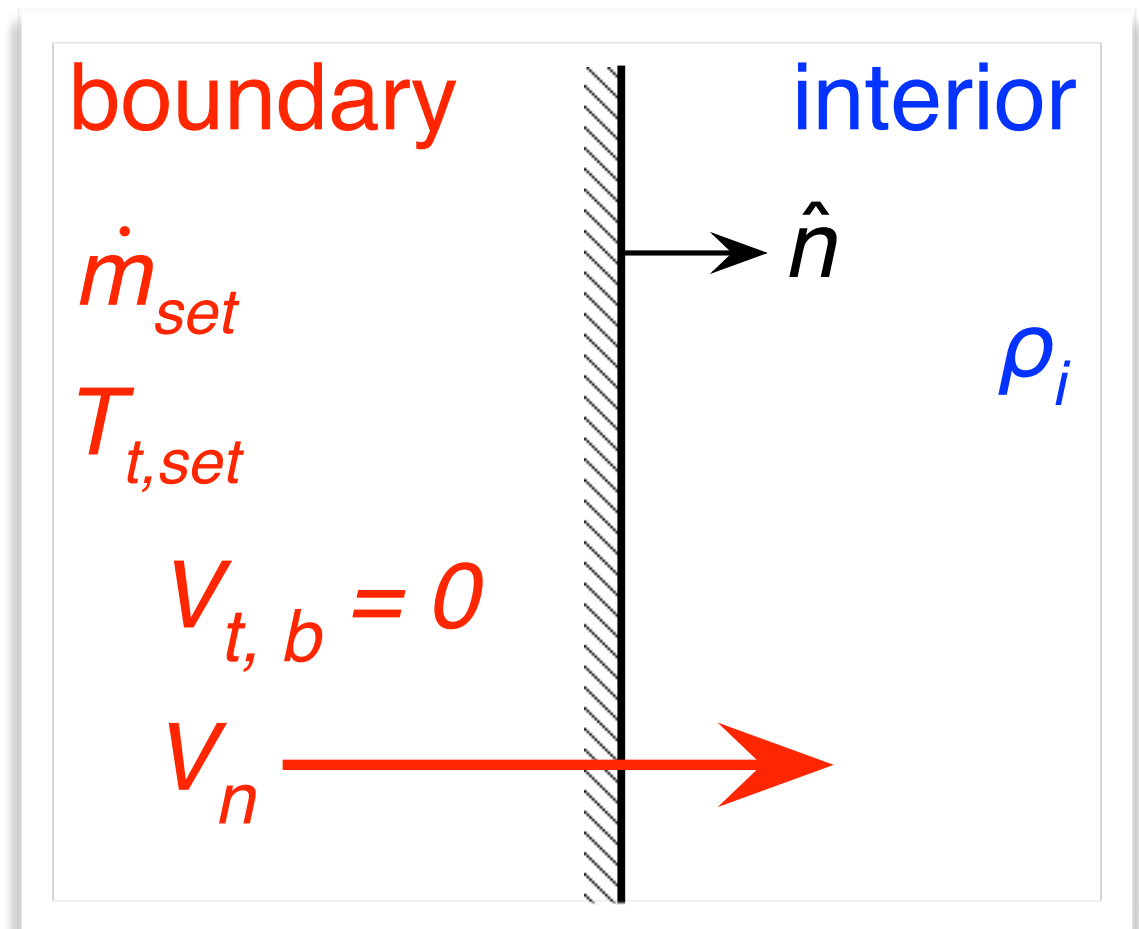
Stagnation Property Inflow

- Most other CFD solvers have this common option

- Total pressure and temperature set to specified value at boundary

- Tangential velocity set to zero, forcing inflow to be normal to surface

- Enthalpy is extrapolated from interior

- Riemann invariant used to computed boundary state

- Safeguards

  - when flow tries to reverse back into boundary, solid wall boundary enforced

  - inflow Mach number is limited to sonic, adjusting stagnation properties accordingly

- Cannot explicitly set a mass flow rate

boundary   interior

$\hat{n}$

$p_{t,set}$

$H_{t,i}$

$T_{t,set}$

J^-
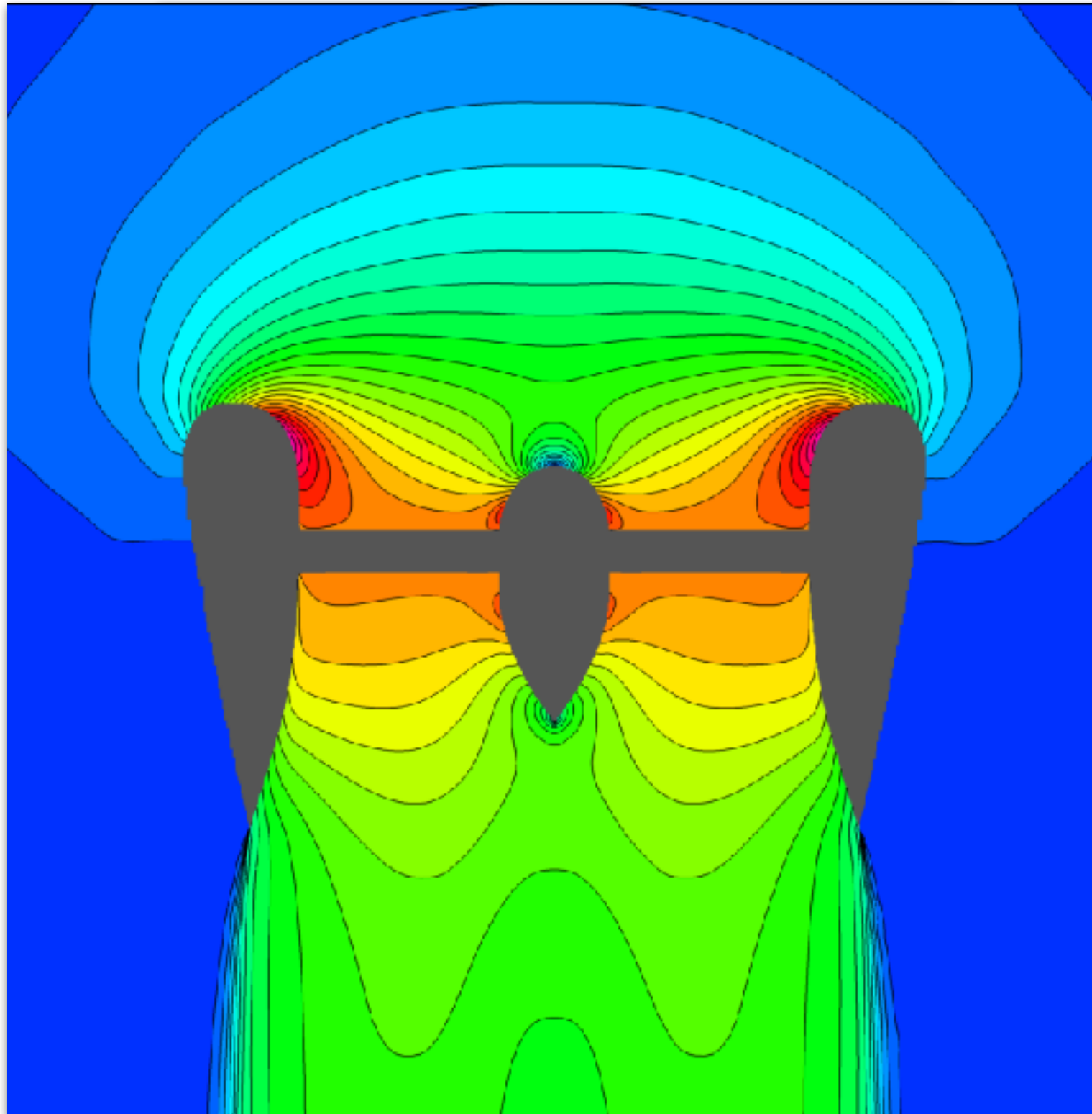
$V_{t,\,b} = 0$

V_n
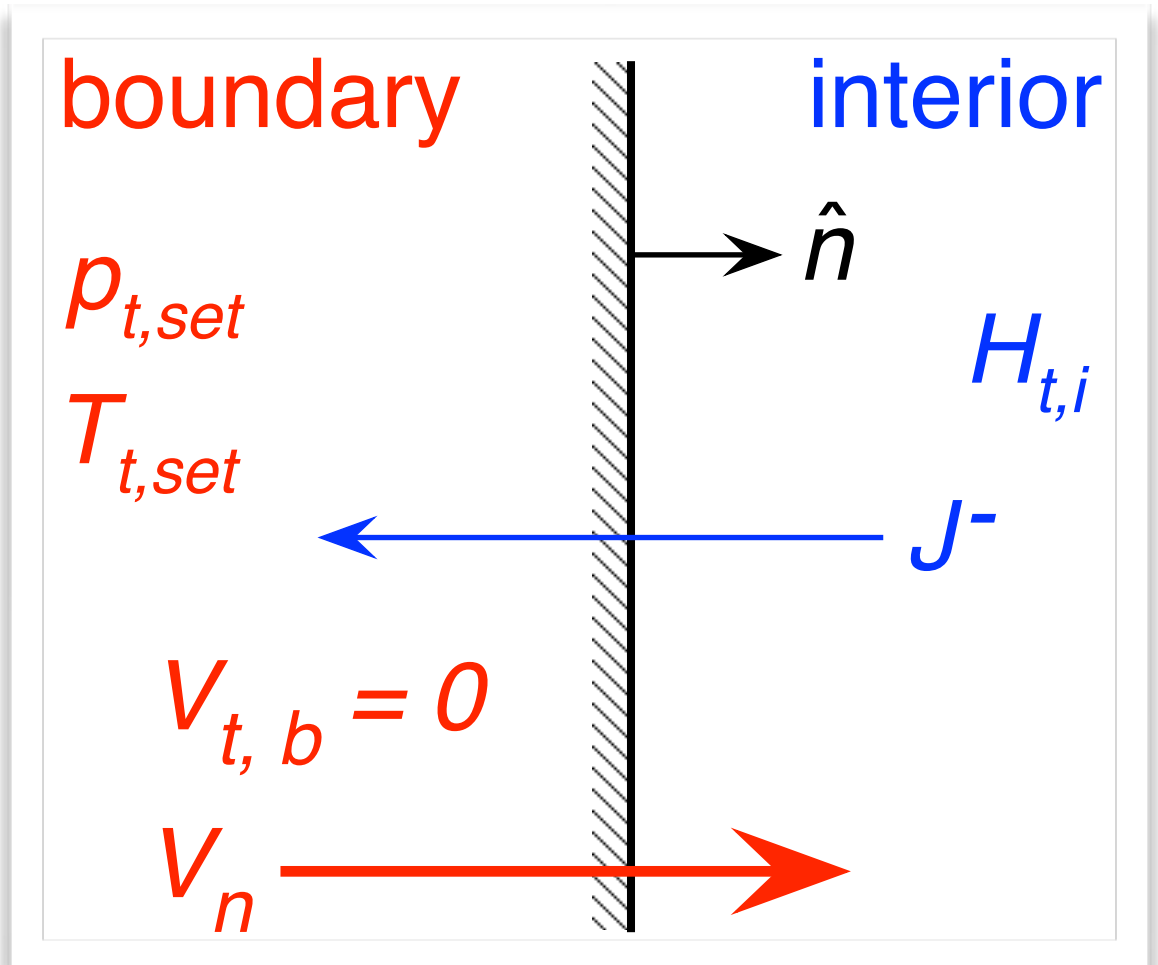
Mass Flow Rate and Total Temperature Inflow

- Allows for explicit mass flow rate control

- Mass flow rate and total temperature set to specified value at boundary

- Tangential velocity set to zero, forcing inflow to be normal to surface

- Density is extrapolated from interior

- Boundary flux computed from boundary state

- Safeguard

 - inflow Mach number is limited to sonic, adjusting boundary values accordingly

boundary   interior

$\dot{m}_{set}$

$T_{t,set}$

$\hat{n}$

ρ_i

$V_{t,\,b} = 0$

V_n

Mass Flow Rate Control

Mass Flow Rate Control

- Constant velocity outflow boundary condition can be steered to obtain specified mass flow rate out of the domain

  - average density over surface is computed

  - velocity out of domain is set based on desired mass flow rate

  - repeat every few iterations until solution converged and mass flow rate within tolerance

$\dot{m}_{outflow}$ **is steered**
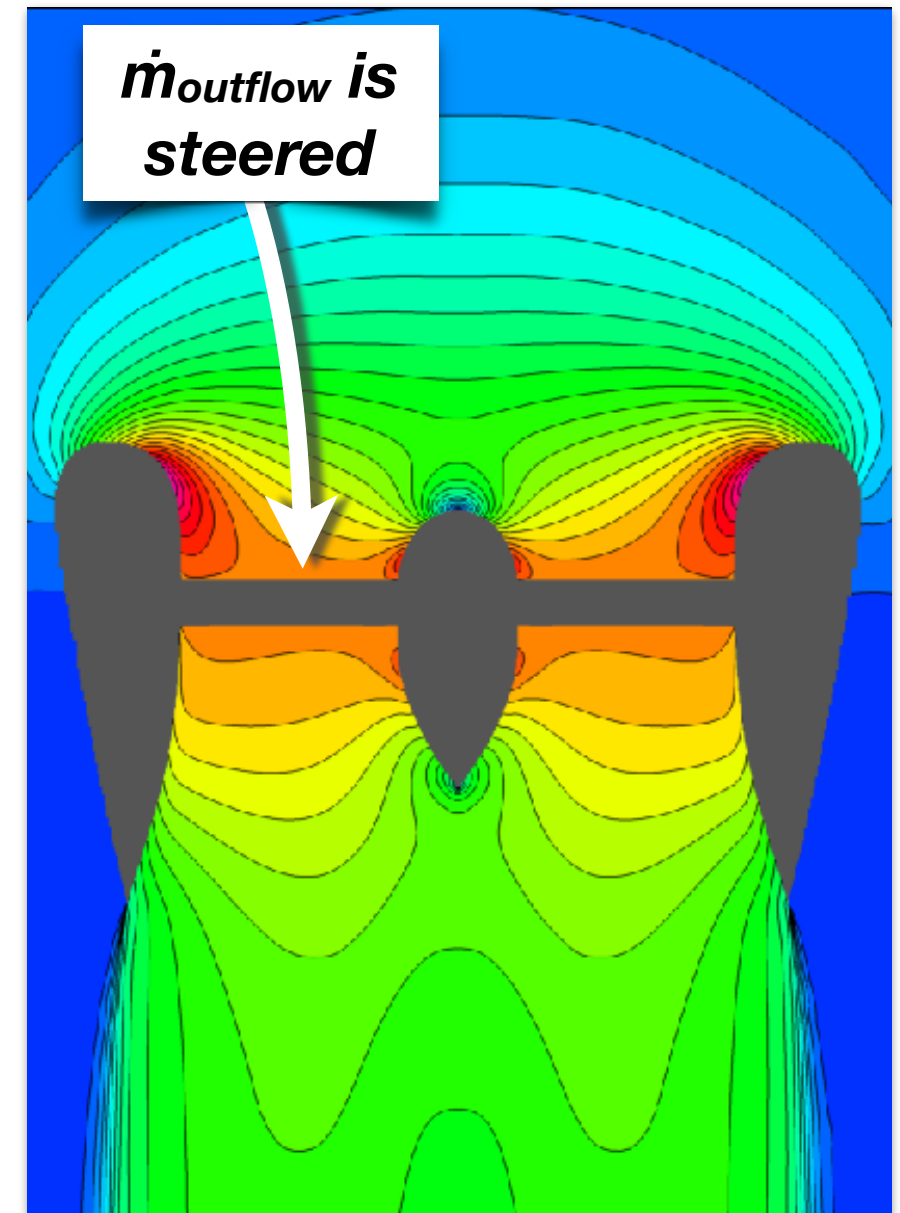
Mass Flow Rate Control

- Constant velocity outflow boundary condition can be steered to obtain specified mass flow rate out of the domain

  - average density over surface is computed

  - velocity out of domain is set based on desired mass flow rate

  - repeat every few iterations until solution converged and mass flow rate within tolerance

- Constant mass flow rate inflow boundary condition explicitly sets mass flow rate into the domain
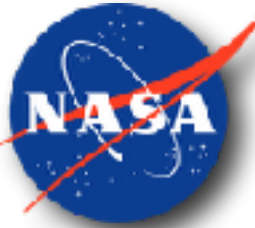

$\dot{m}_{inflow}$ **is set**
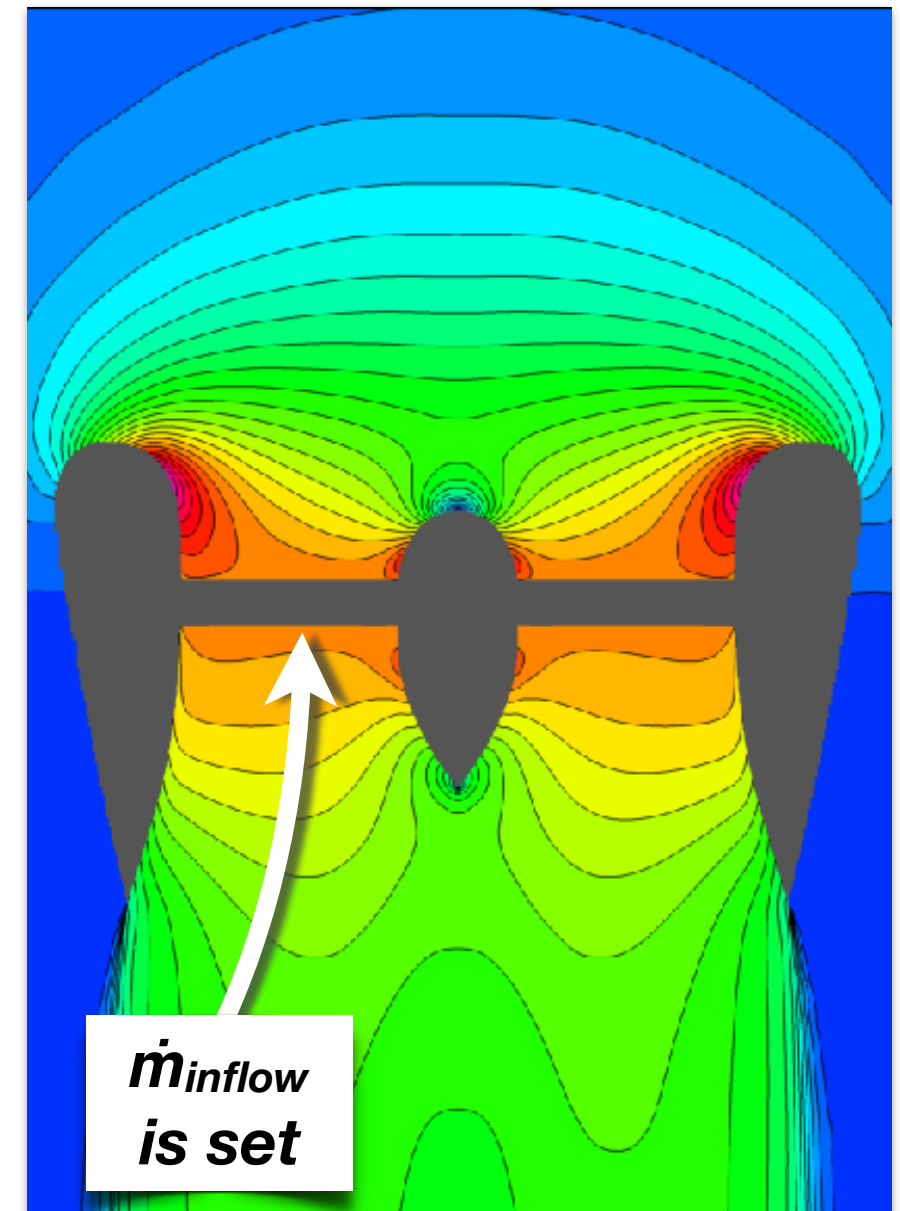
Adjoint-Driven Adaptive Mesh Refinement

- All boundary conditions now implemented in adaptive mesh refinement process

- Updates to **adjointCart**, **xSensit**, **adjointErrorEstQuad**, etc.

adapt 5

adapt 7

adapt 9

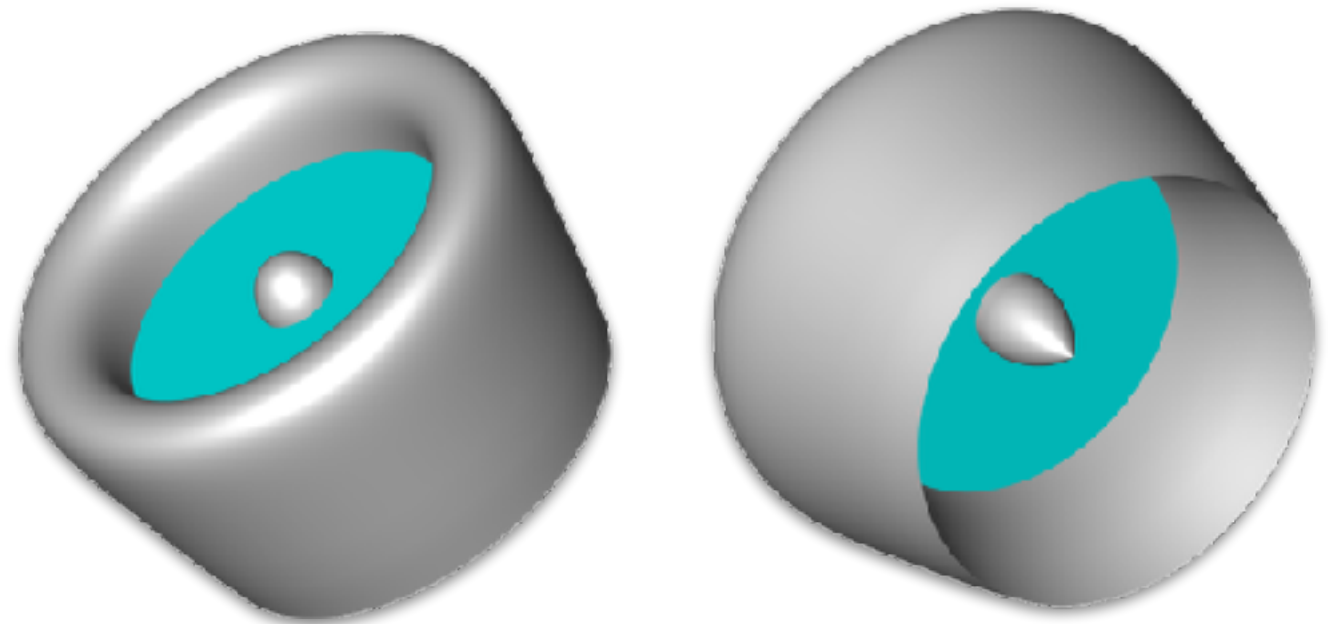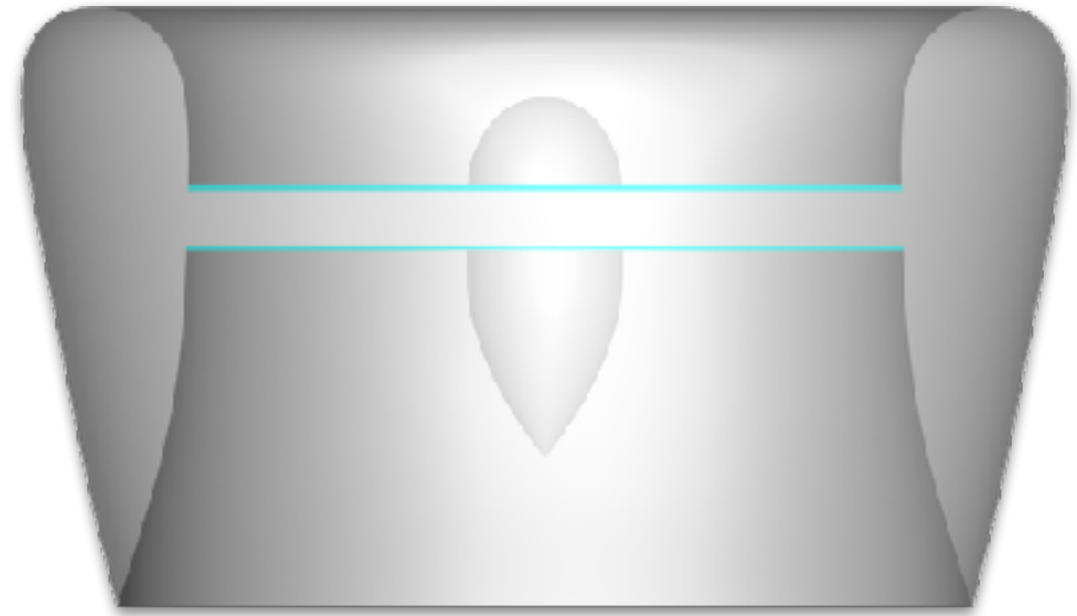adapt 11
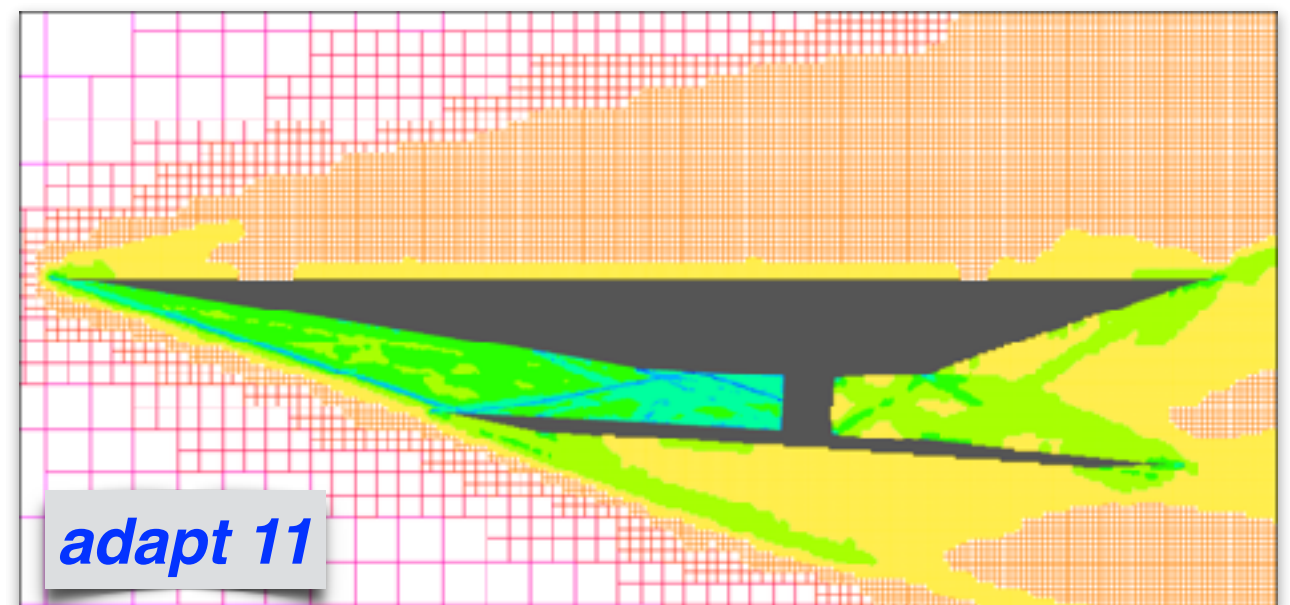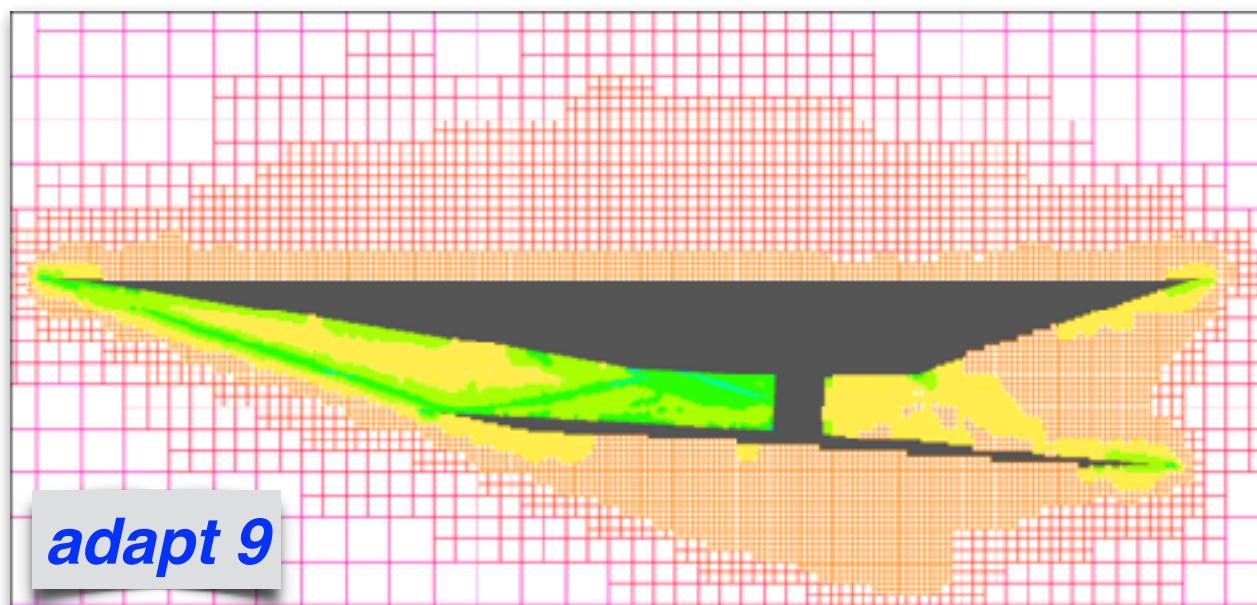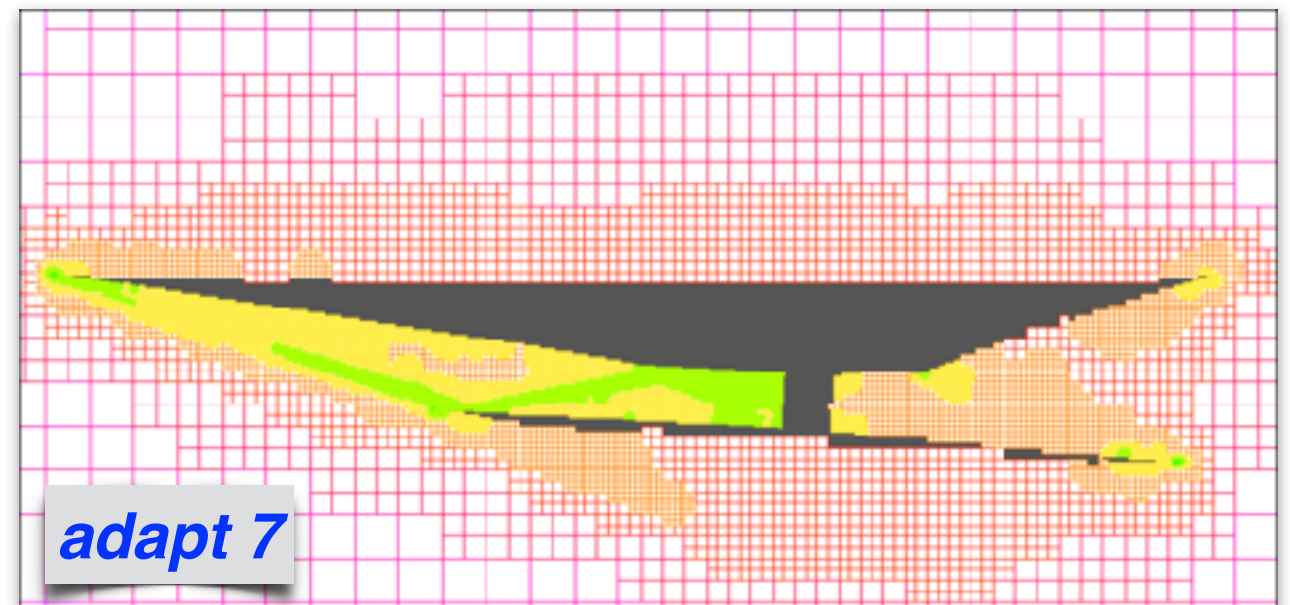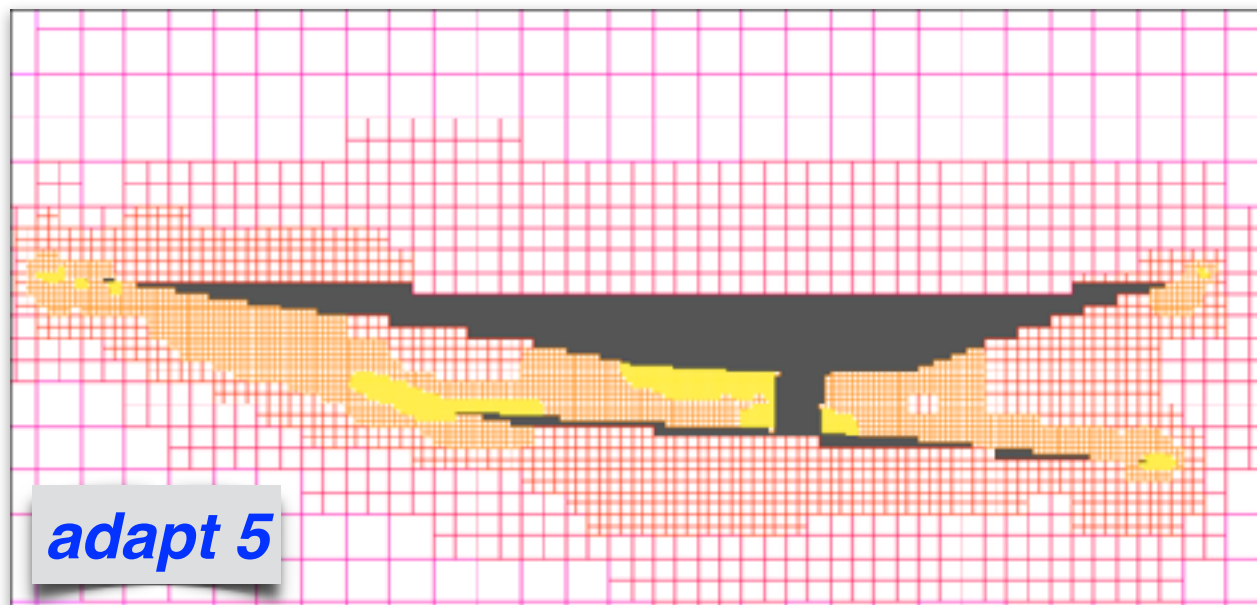
Application of New Boundary Conditions

- Ducted fan in near-hover (subsonic)

    - verification of back pressure outflow and mass flow rate inflow boundary conditions

- Turbofan with both fan and turbine exhaust streams (transonic)

    - verification of constant velocity outflow boundary condition

- Turbojet with 2-D ramp inlet (supersonic)

    - mesh convergence through adaptive refinement

    - mass flow rate steering example

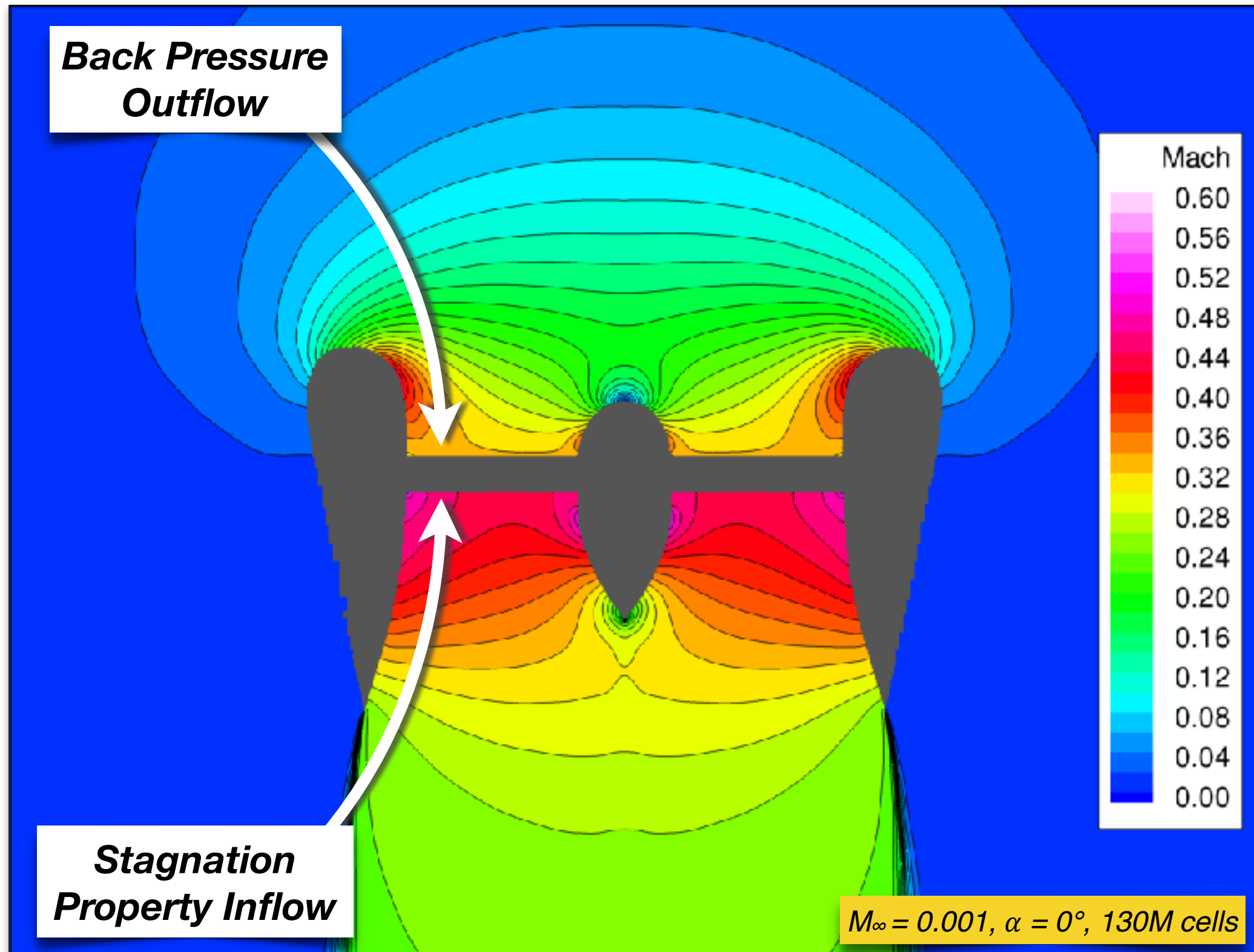- Scramjet (hypersonic)

- Low boom demonstrator
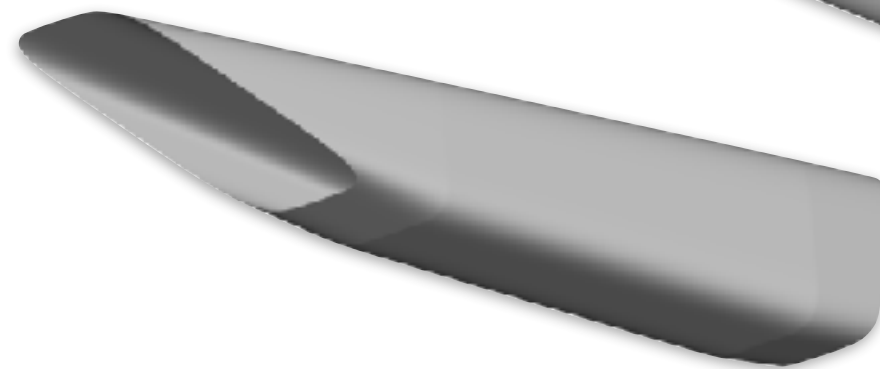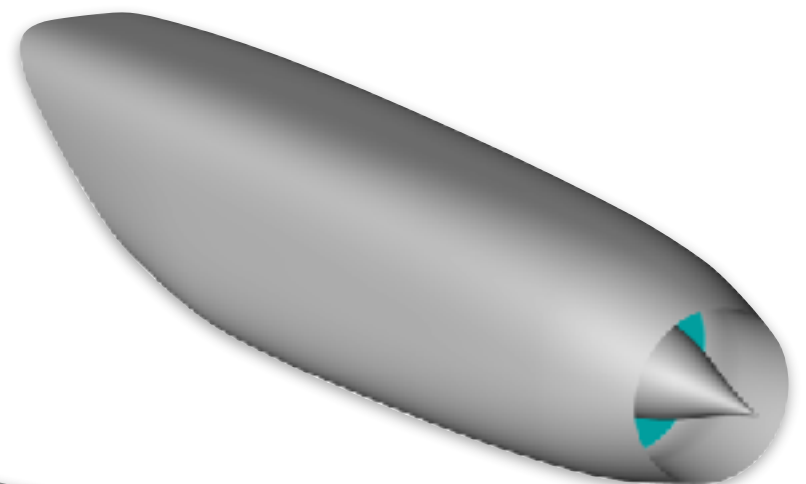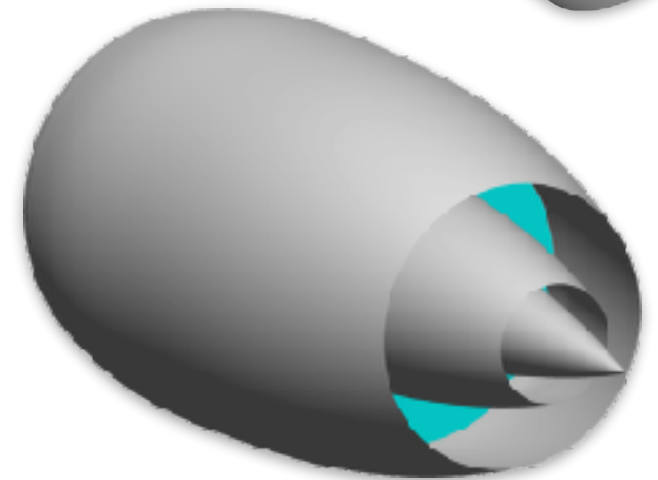
Ducted Fan in Hover

- Duct and center body housing motor to drive fan

- Very low freestream Mach number (0.001) to simulate near hover

- No angle of attack - axisymmetric flow

- Fan modeled as **annular disk**

- Inflow / Outflow boundary conditions enforced on disk to model fan effects
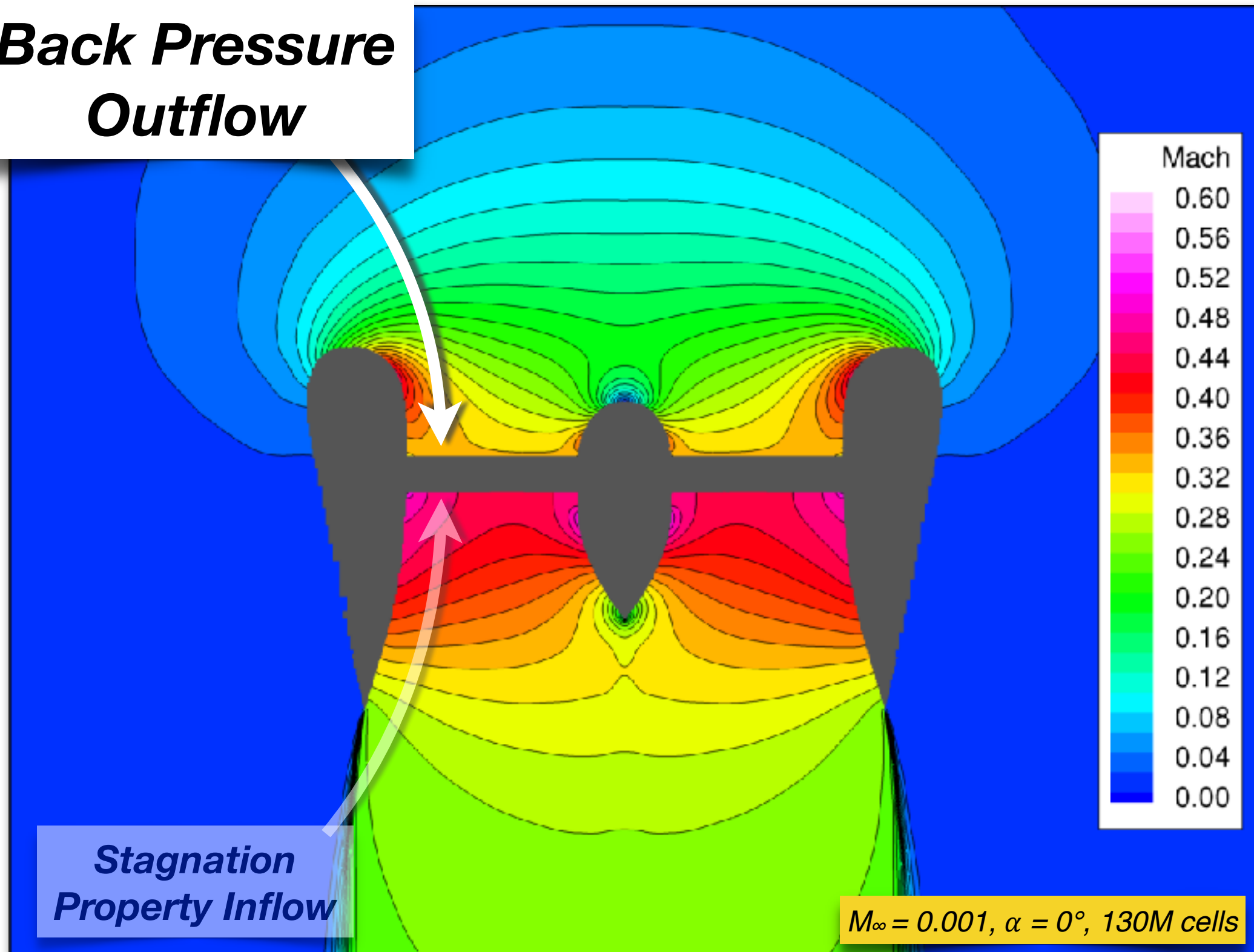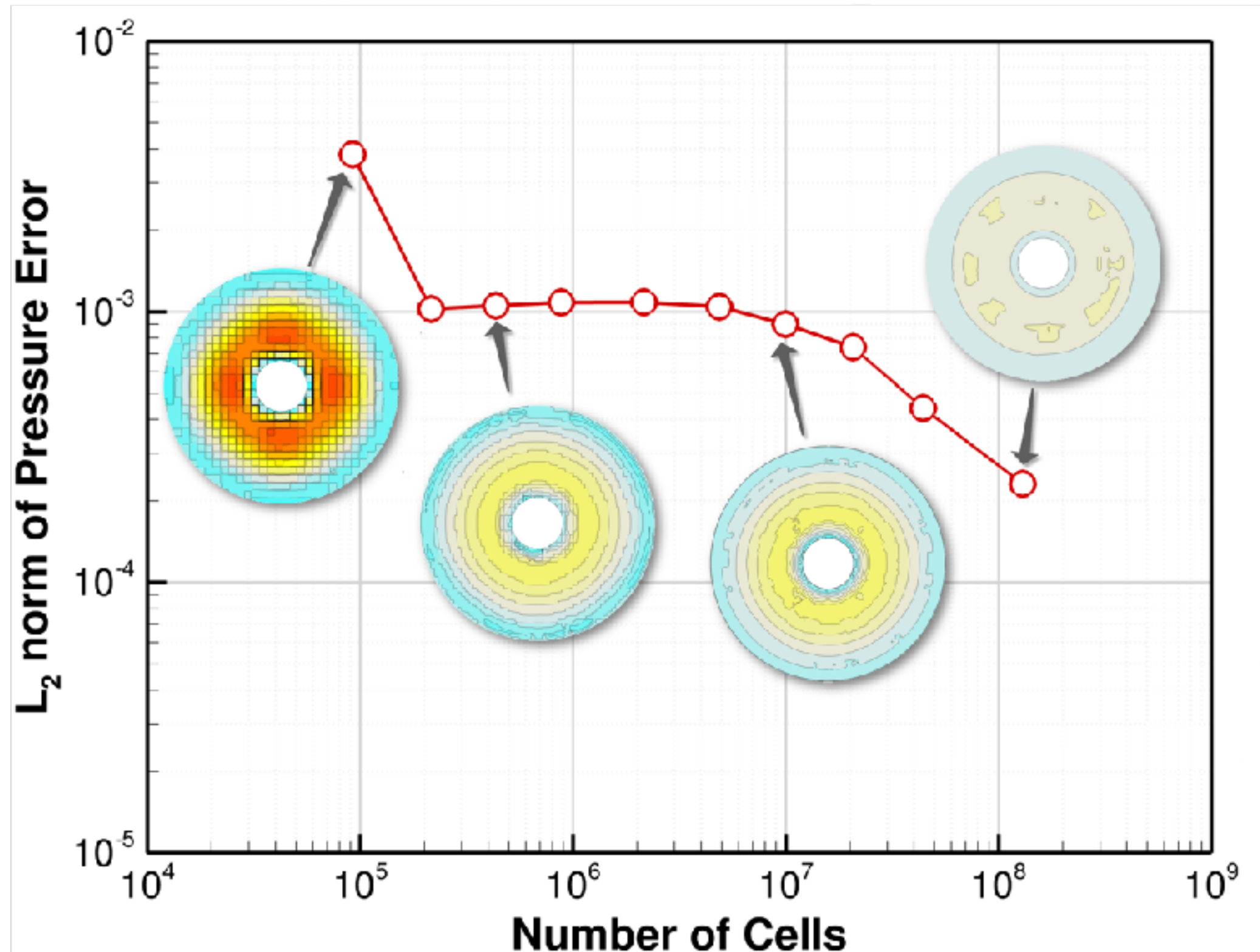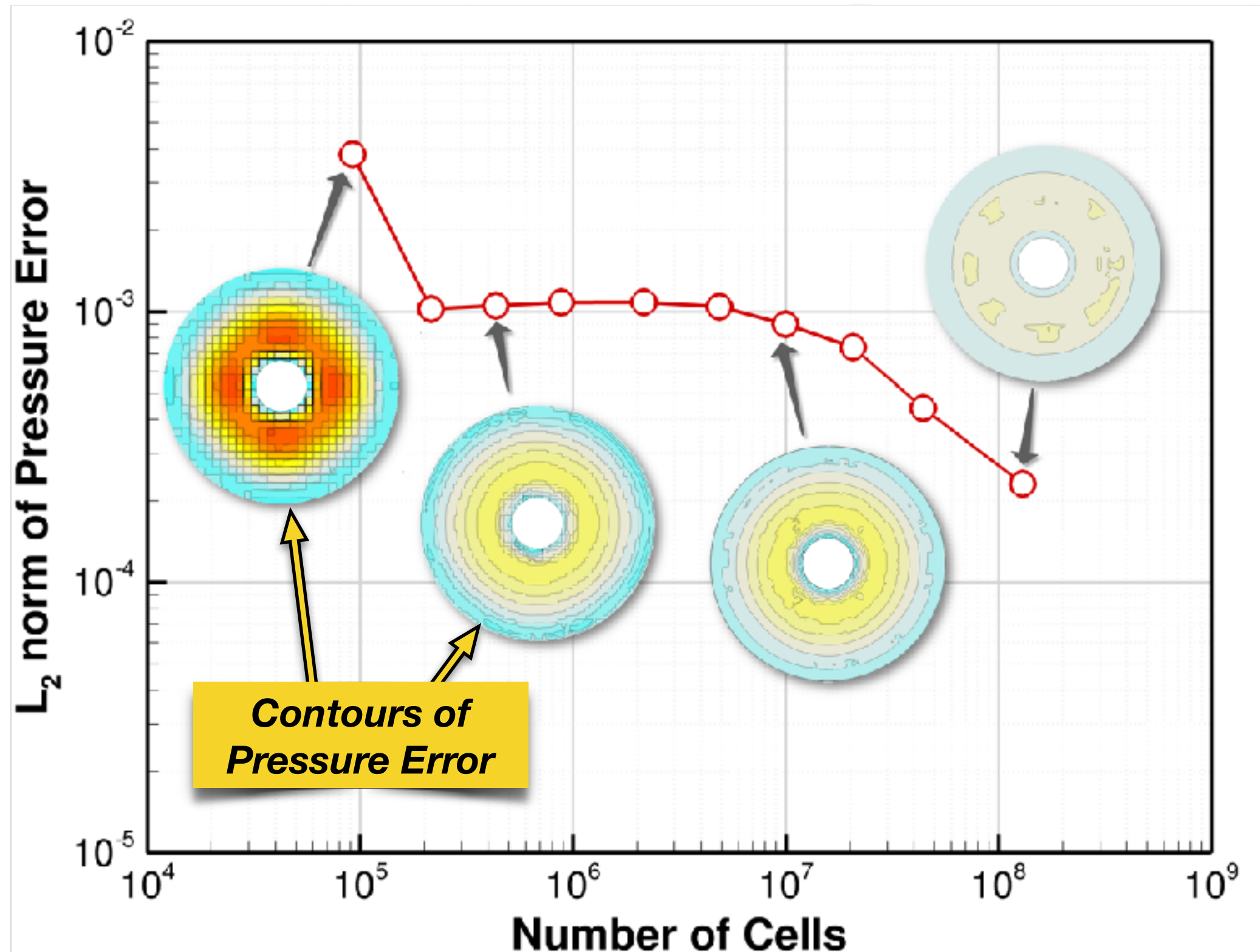
Ducted Fan - Example Solution

Back Pressure Outflow

Stagnation Property Inflow

Mach
0.60
0.56
0.52
0.48
0.44
0.40
0.36
0.32
0.28
0.24
0.20
0.16
0.12
0.08
0.04
0.00

$M_\infty = 0.001,\ \alpha = 0°,\ 130M\ cells$

Ducted Fan - Example Solution

Back Pressure Outflow

Stagnation Property Inflow

Mach
0.60
0.56
0.52
0.48
0.44
0.40
0.36
0.32
0.28
0.24
0.20
0.16
0.12
0.08
0.04
0.00

$M_\infty = 0.001$, $\alpha = 0°$, 130M cells

Back Pressure B.C. Mesh Convergence

Back Pressure B.C. Mesh Convergence

Contours of Pressure Error

Ducted Fan - Example Solution

Constant Velocity Outflow

Constant Mass Flow Rate Inflow

Mach
0.60
0.56
0.52
0.48
0.44
0.40
0.36
0.32
0.28
0.24
0.20
0.16
0.12
0.08
0.04
0.00

$M_\infty = 0.001$, $\alpha = 0°$, 160M cells

Ducted Fan - Example Solution

Constant Velocity Outflow

Constant Mass Flow Rate Inflow

$M_\infty = 0.001$, $\alpha = 0°$, 160M cells

Mach
- 0.60
- 0.56
- 0.52
- 0.48
- 0.44
- 0.40
- 0.36
- 0.32
- 0.28
- 0.24
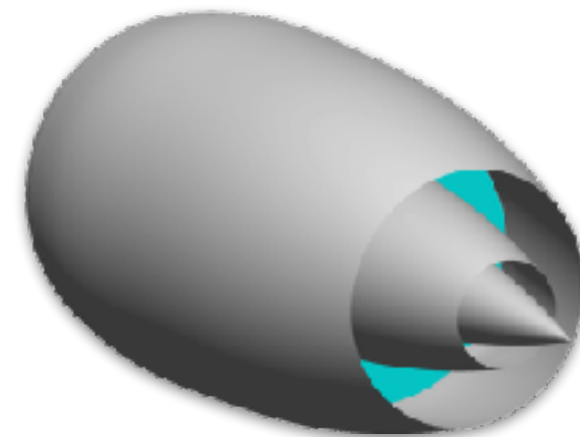- 0.20
- 0.16
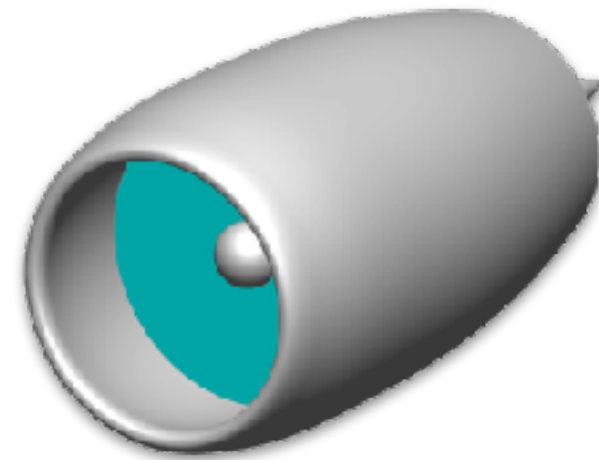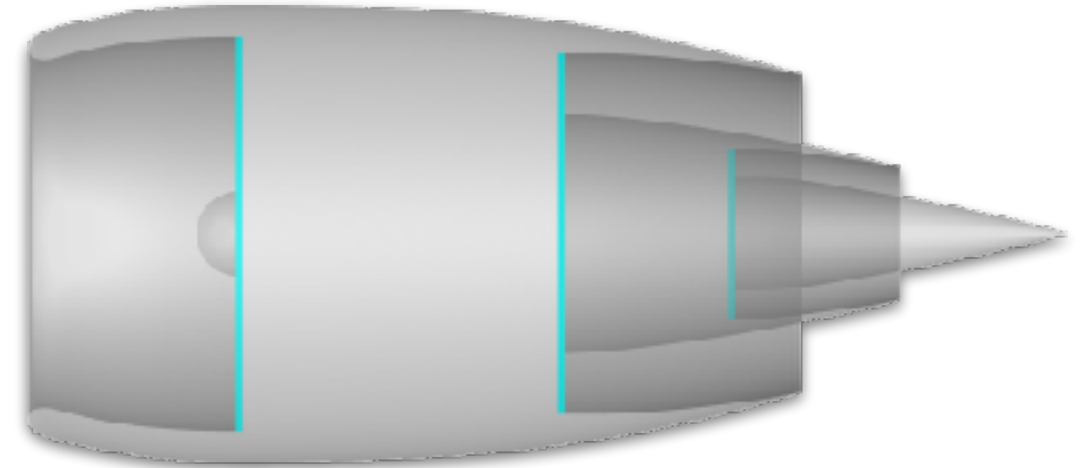- 0.12
- 0.08
- 0.04
- 0.00
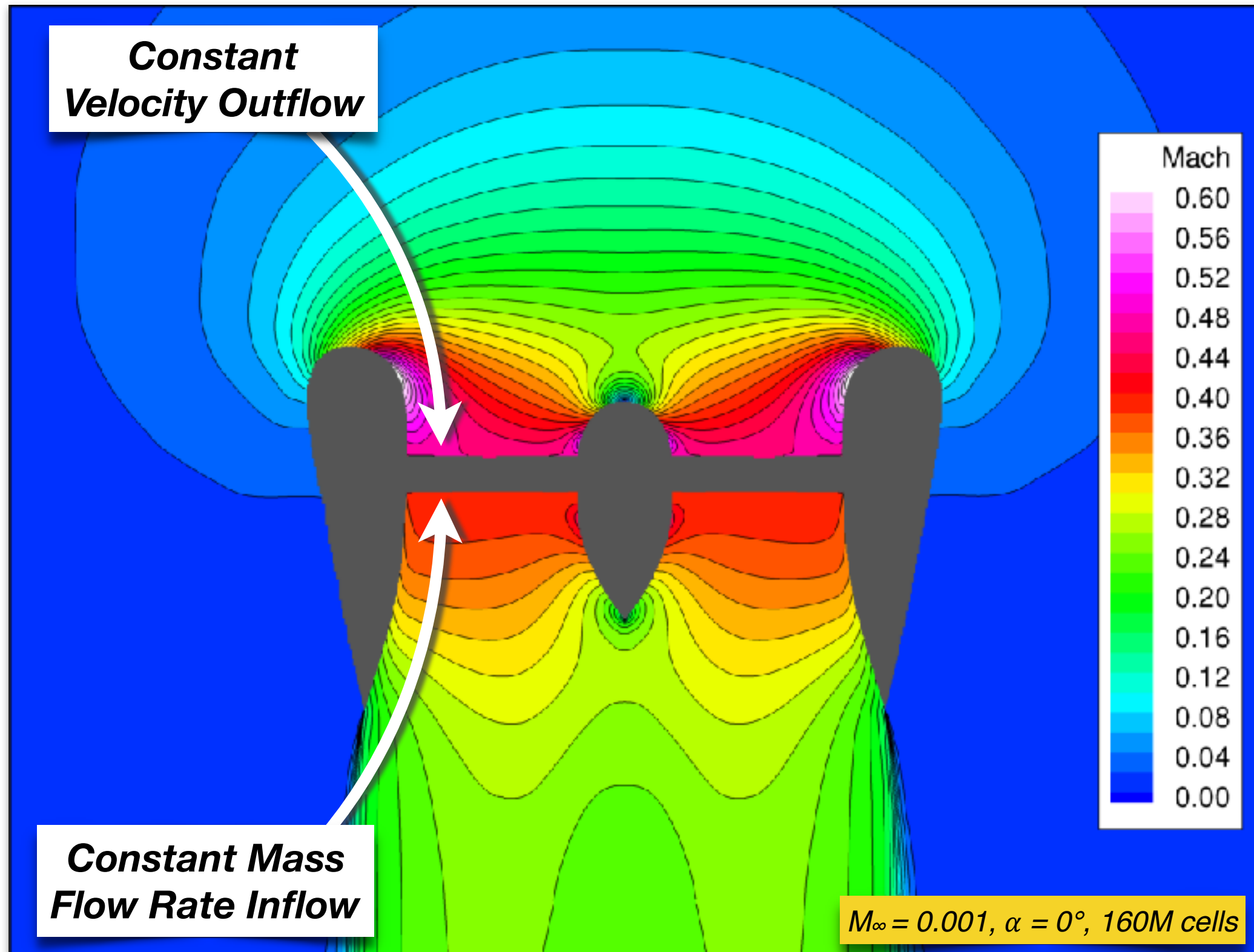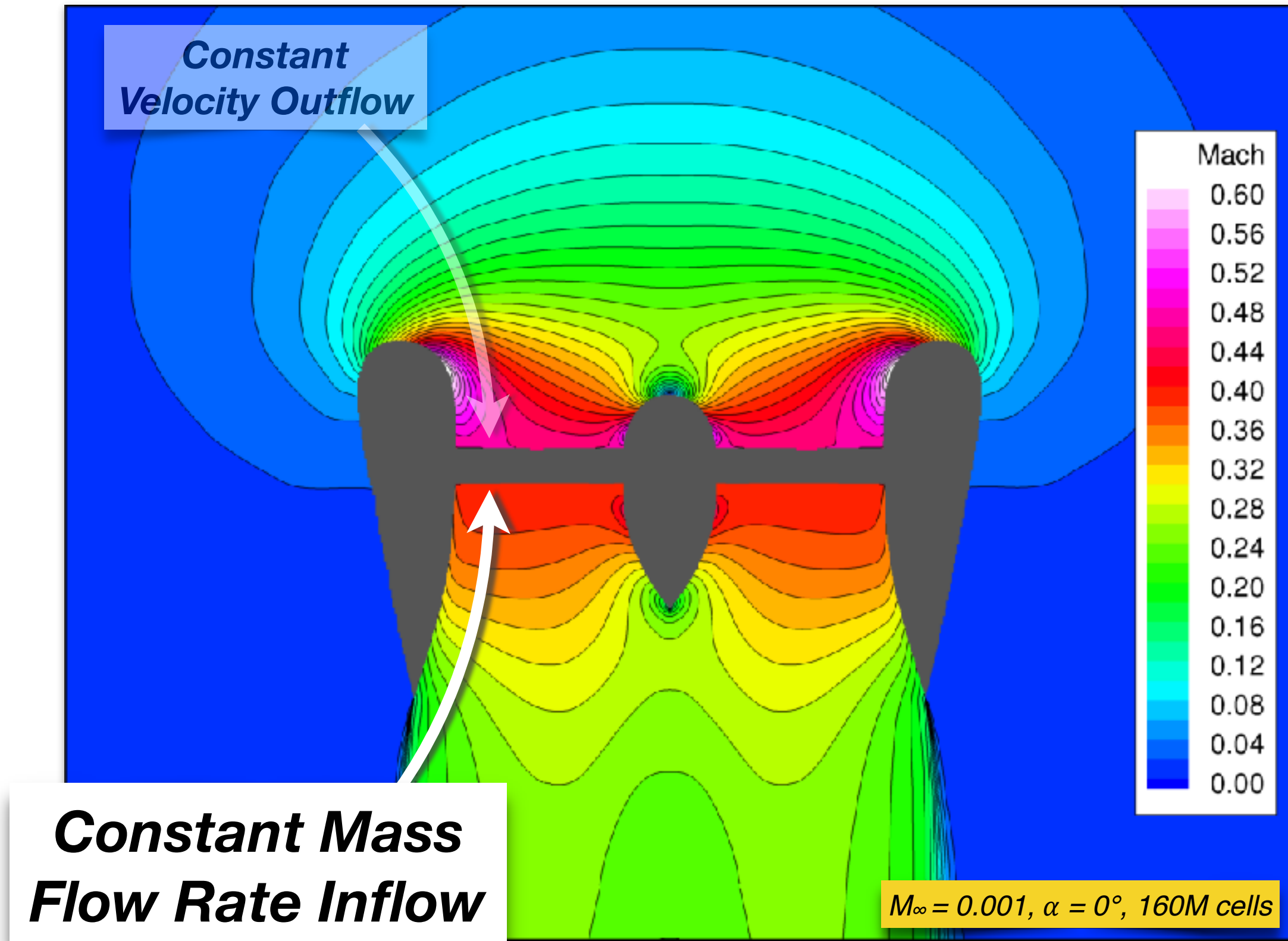
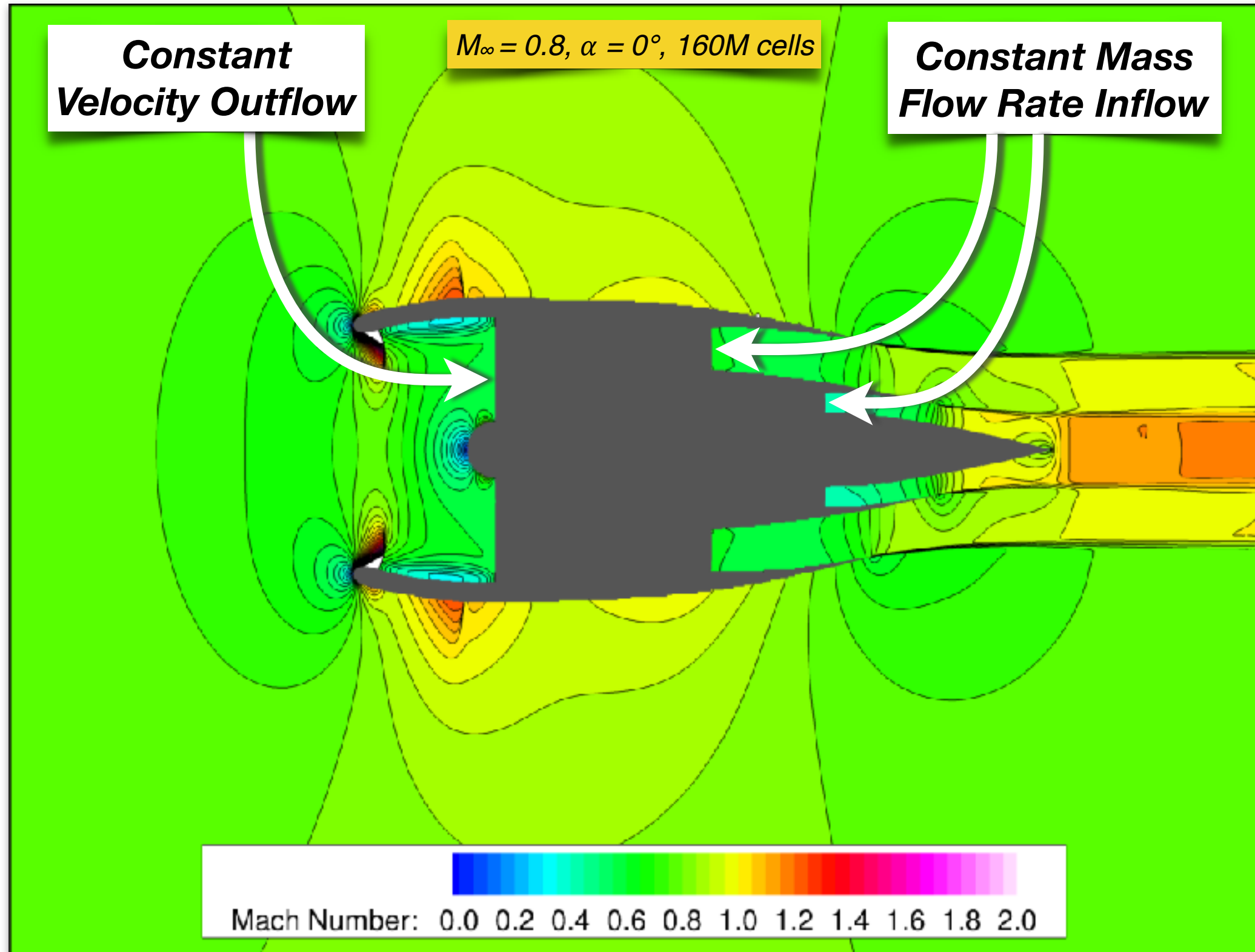# Constant Mass Flow Rate B.C. Mesh Convergence
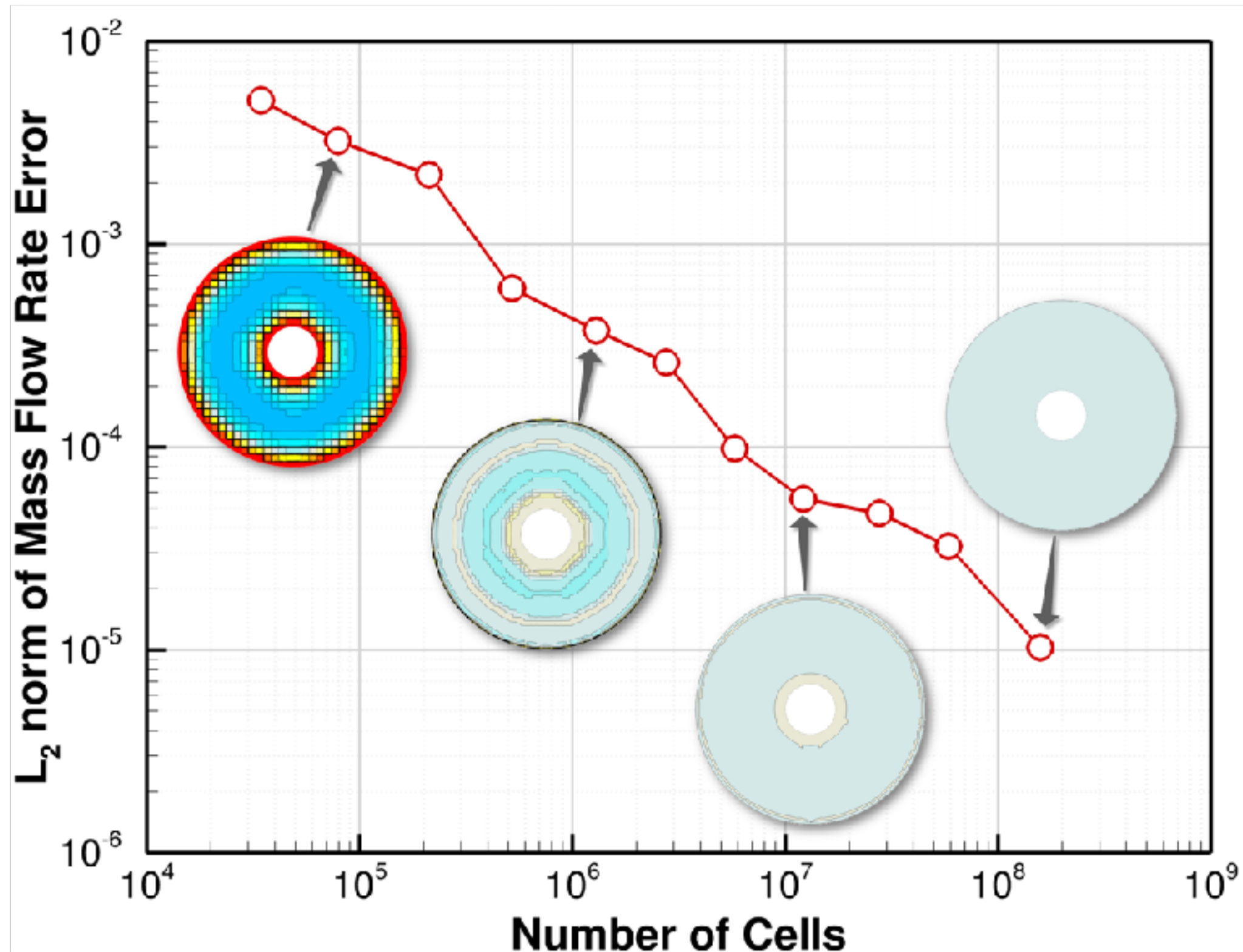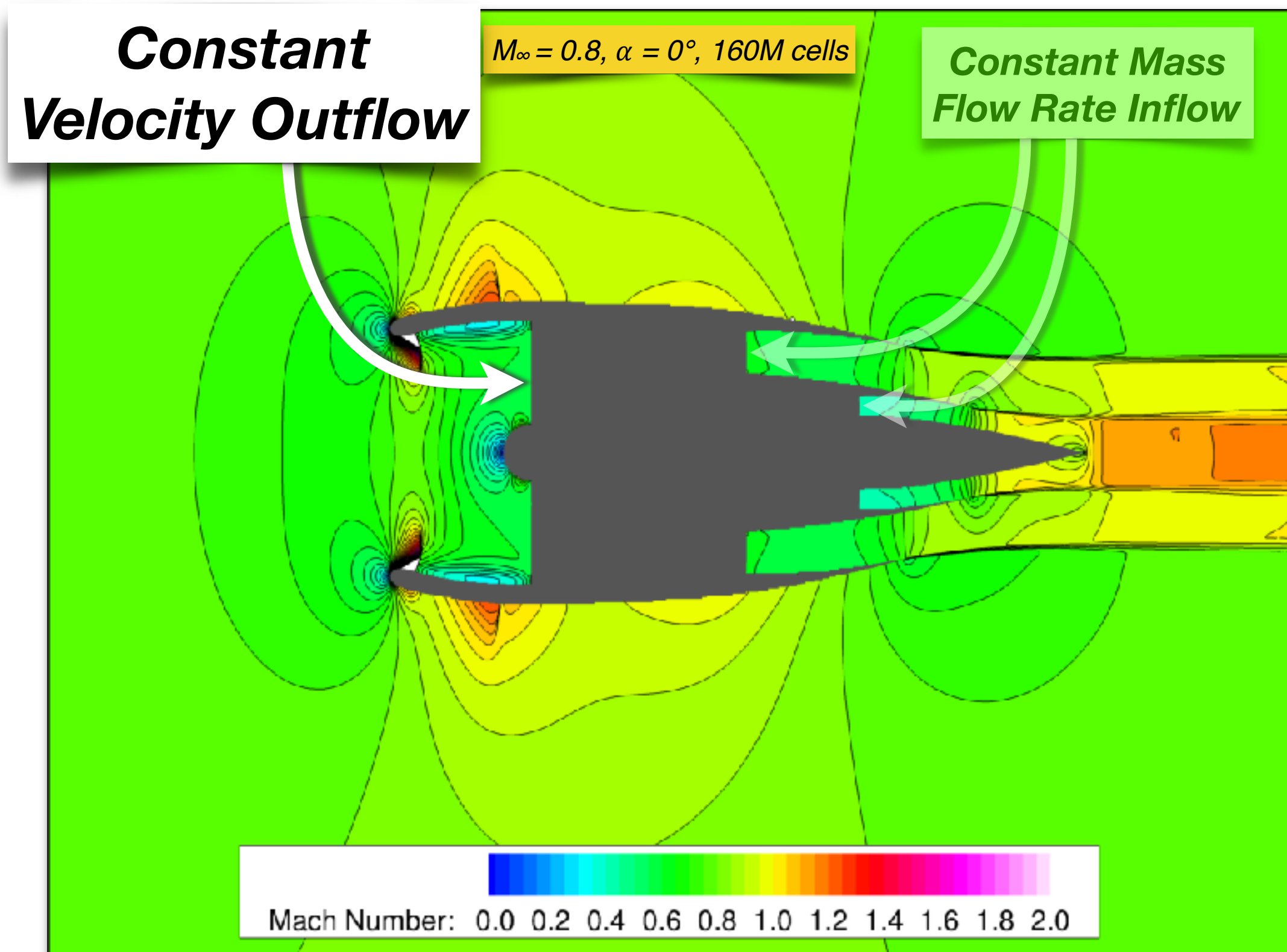
Turbofan in Transonic Flow

- Transonic diffuser with fan hub

- Two stream exhaust with cone nozzle for turbine flow

- Mach 0.8 freestream, no angle of attack (axisymmetric flow)

- Fan / Compressor face modeled as **annulus**, outflow boundary condition applied

- Fan and turbine exhaust planes modeled as **annuli**, inflow boundary conditions applied
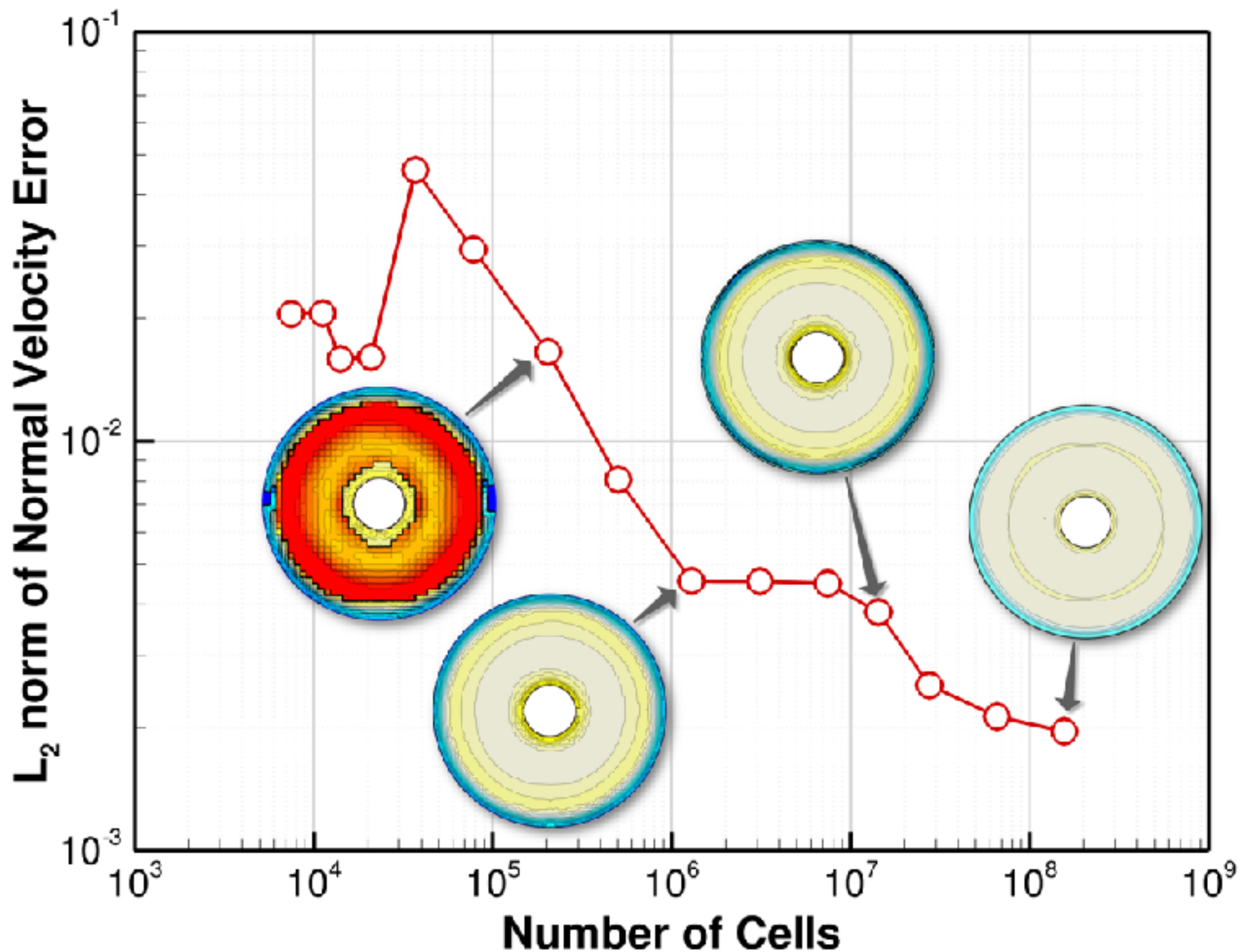
Turbofan - Example Solution

Turbofan - Example Solution

Constant Velocity Outflow

M∞ = 0.8, α = 0°, 160M cells

Constant Mass Flow Rate Inflow

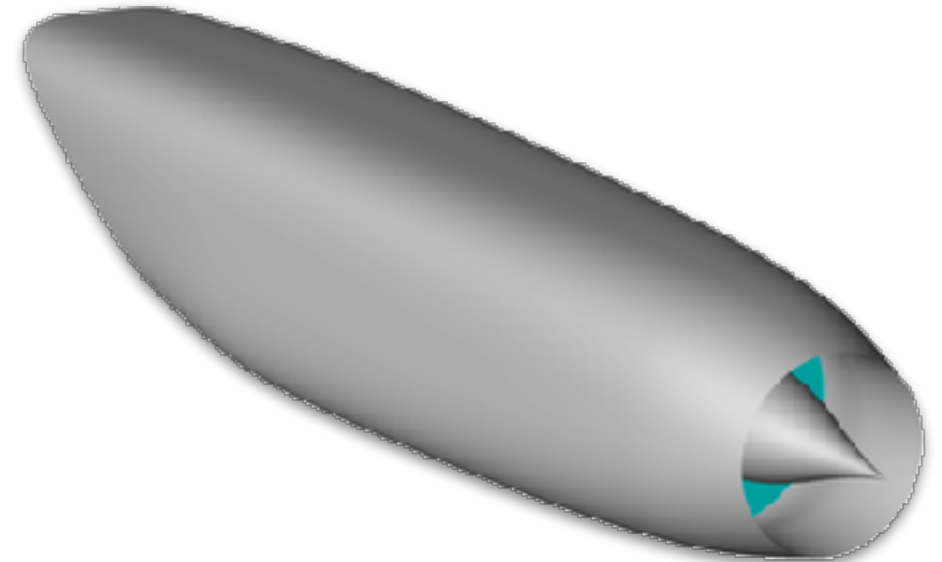Mach Number: 0.0 0.2 0.4 0.6 0.8 1.0 1.2 1.4 1.6 1.8 2.0
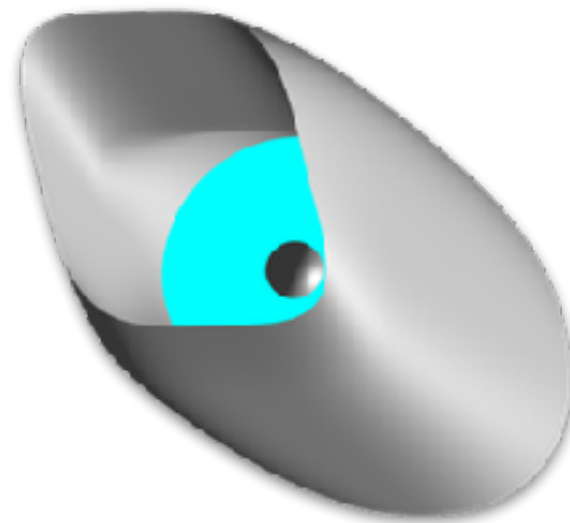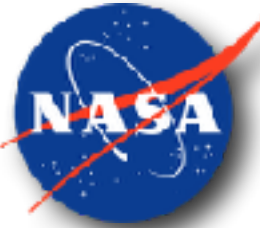
Constant Velocity B.C. Mesh Convergence
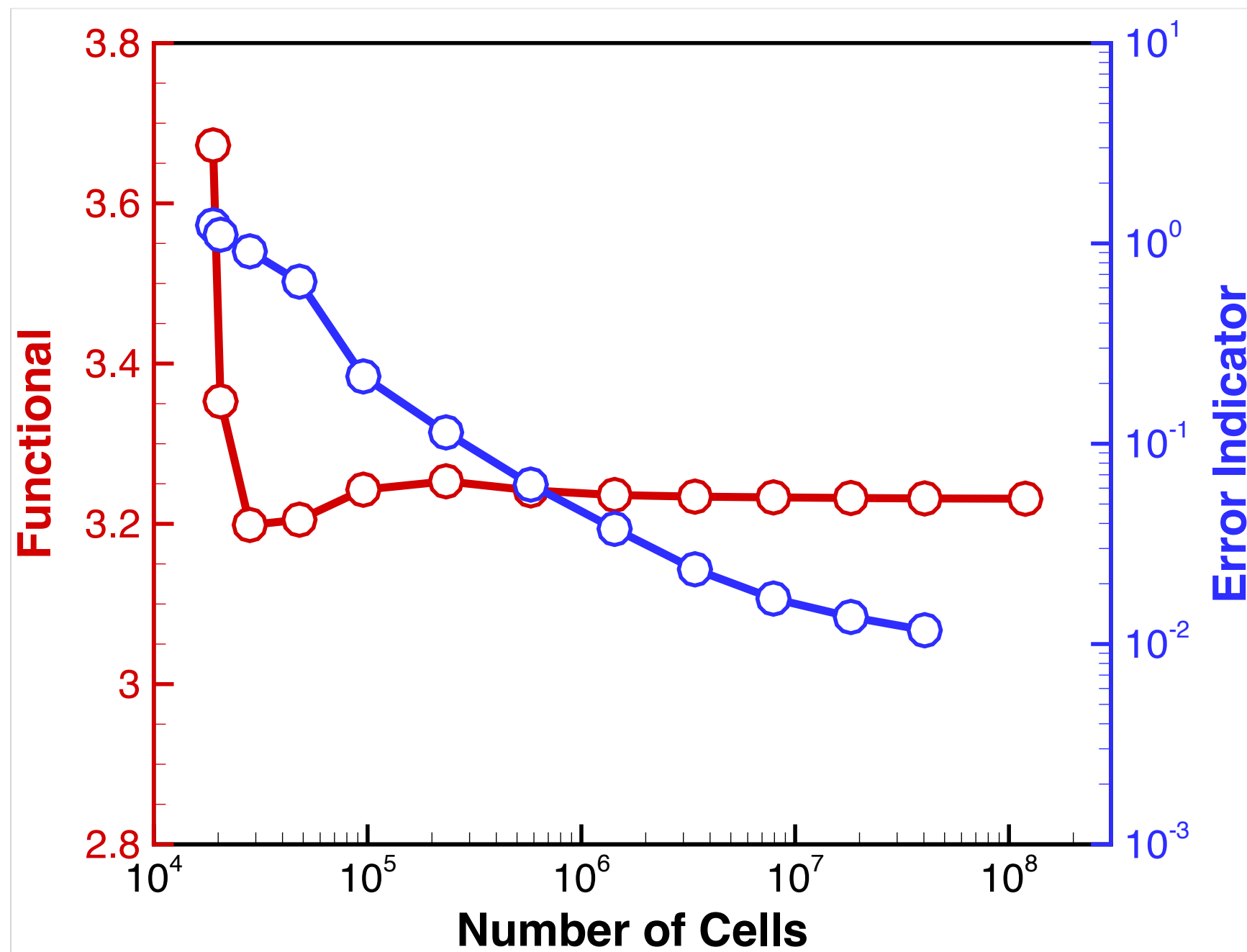
Turbojet in Supersonic Flow

- 2-D ramp inlet design for normal terminal shock

- Converging-diverging duct with cone nozzle

- Mach 1.5 freestream, 1° angle of attack

- Outflow / Inflow boundary conditions applied to **annuli**
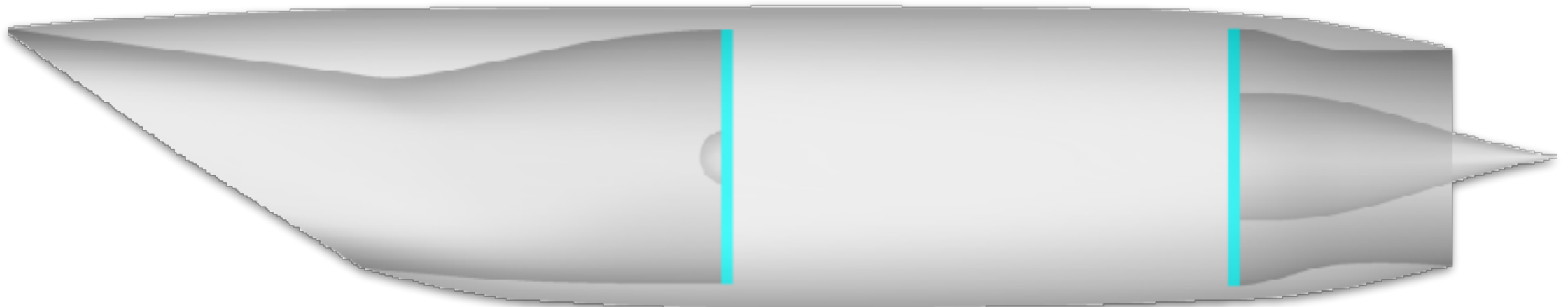
Turbojet - Mesh Convergence

- Good convergence of functional (thrust + lift + plume sensor)

- Steady reduction in error estimate
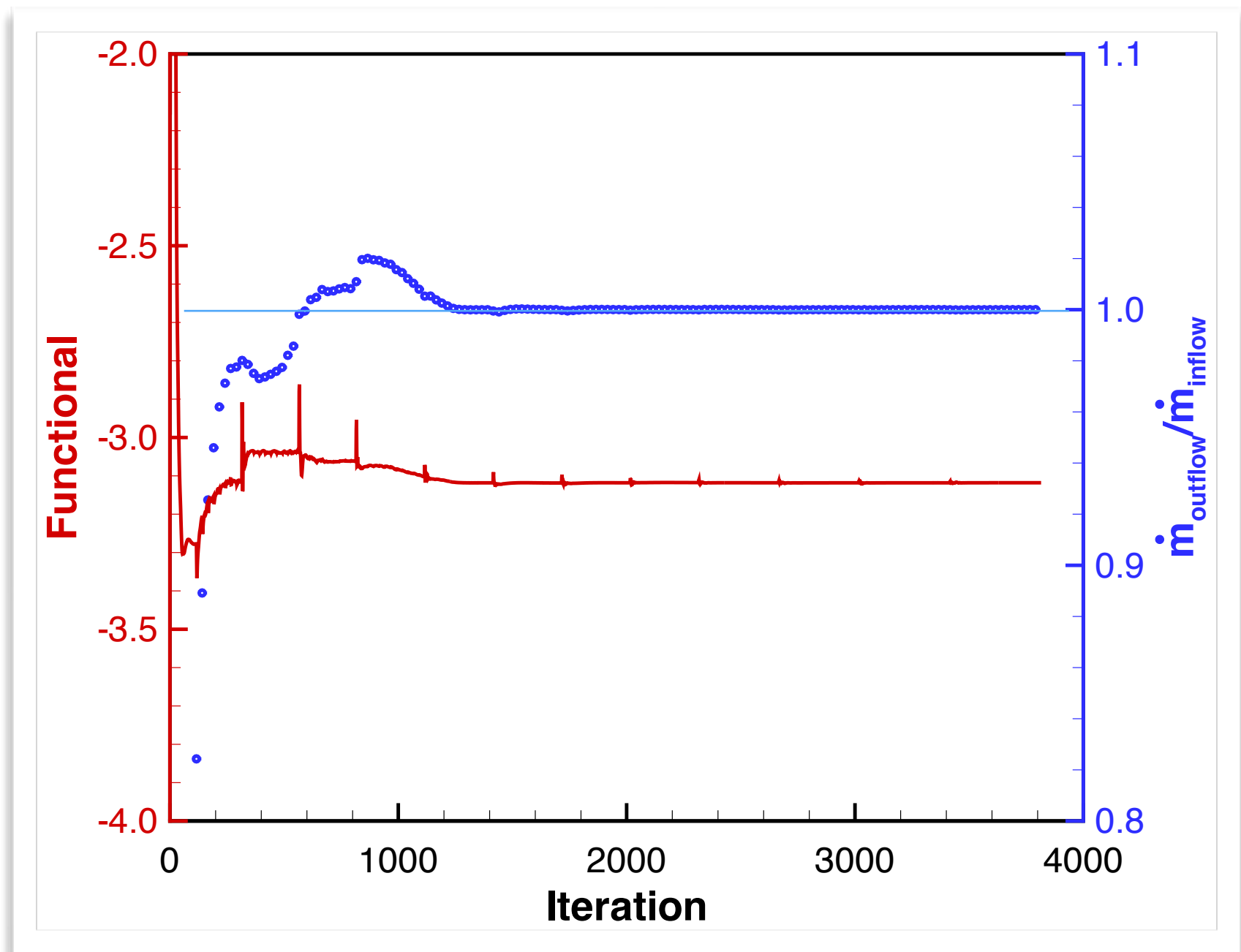
Supersonic Inlet - Mass Flow Rate

- Usually need to specify mass flow rate through an inlet

- Often desirable to match nozzle mass flow rate if modeled

- Highly nonlinear flow features can make mass flow rate steering difficult in supersonic and even transonic inlets
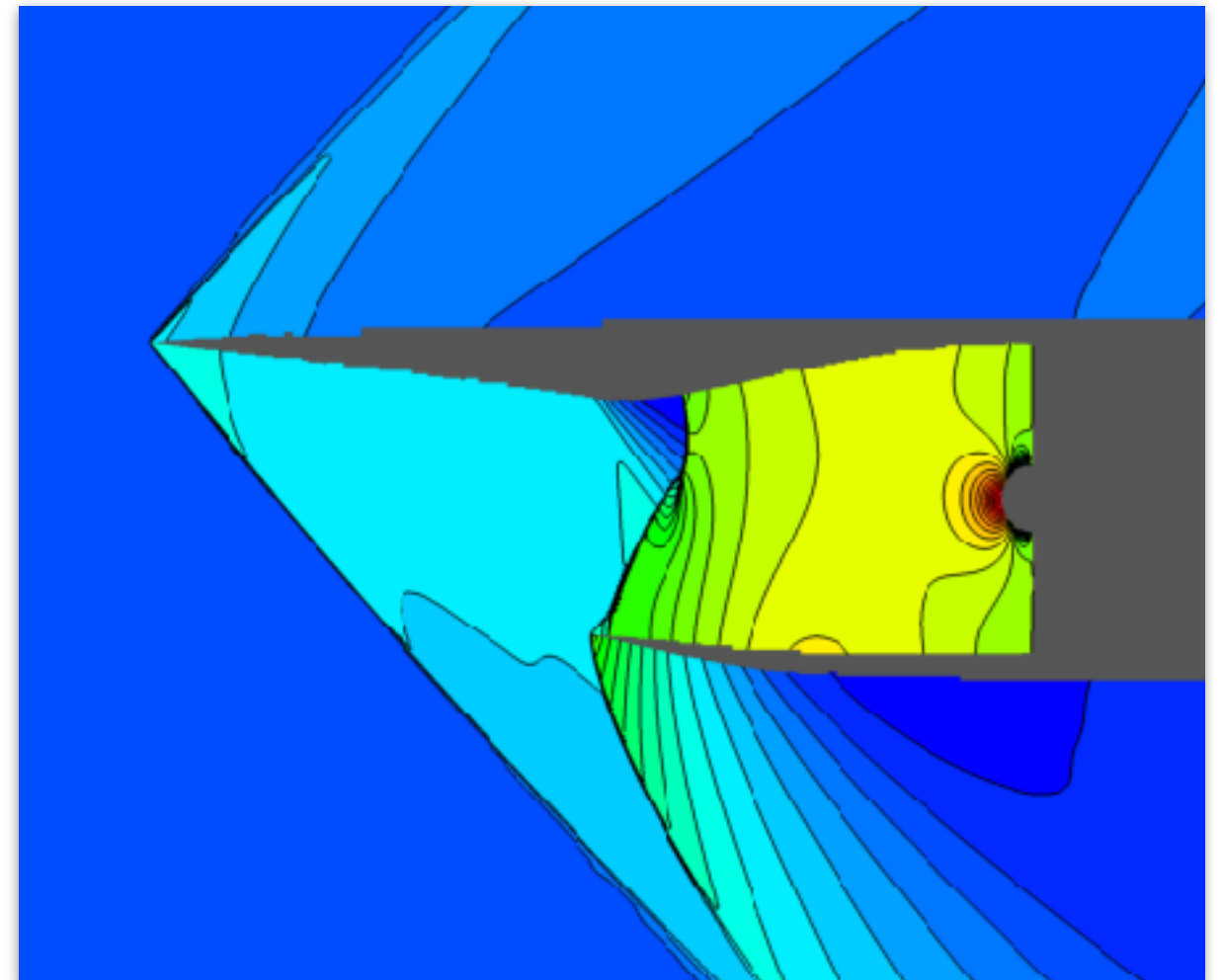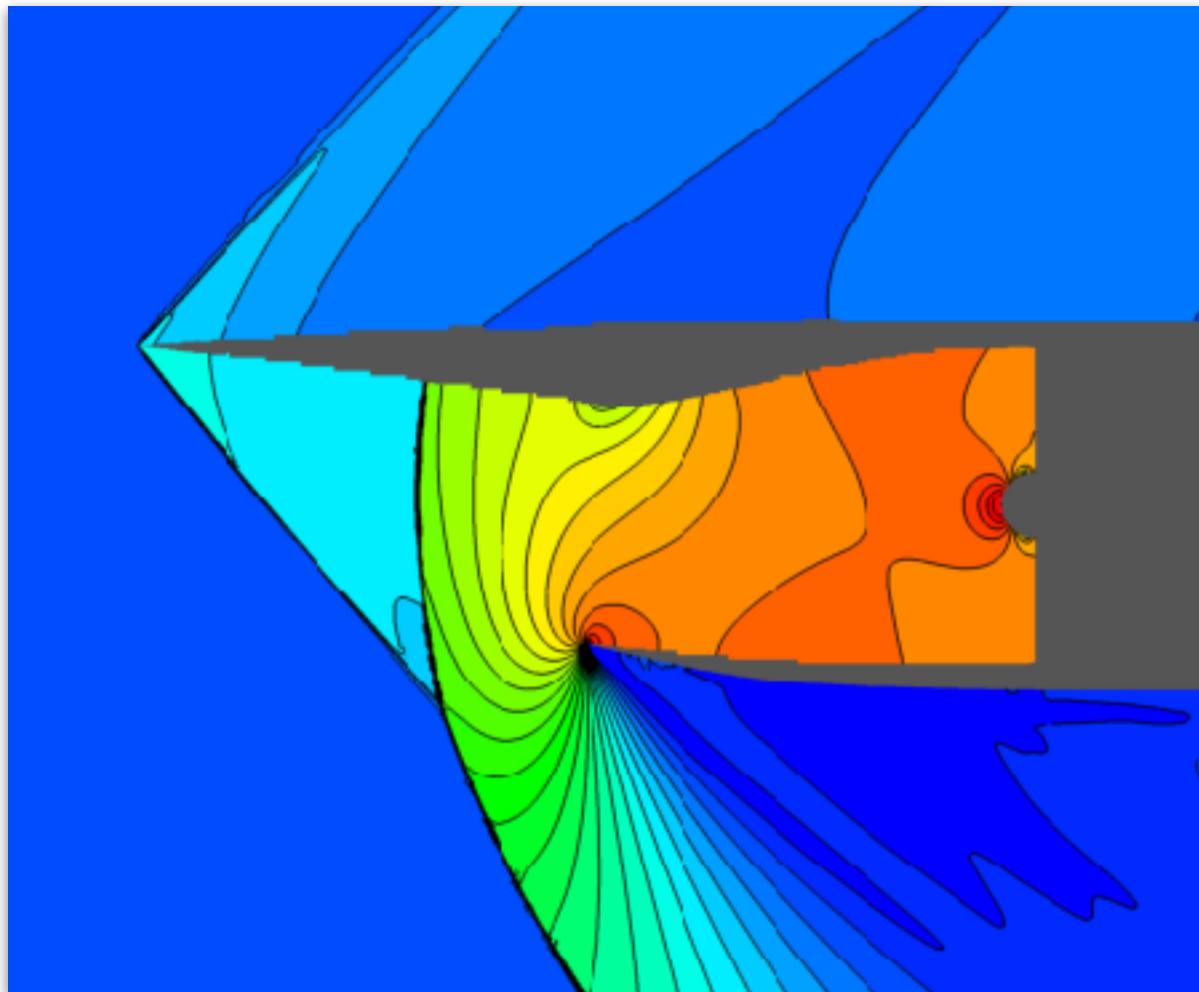
Turbojet - Mass Flow Rate Steering

- Inflow mass flow rate ($\dot{m}_{inflow}$) set through boundary condition

- Outflow mass flow rate ($\dot{m}_{outflow}$) steered to match

- Mass flow rate quickly converges and continues to converge through each refined mesh

Turbojet - Mass Flow Rate Steering

- Inflow mass flow rate ($\dot{m}_{inflow}$) set through boundary condition

- Outflow mass flow rate ($\dot{m}_{outflow}$) steered to match

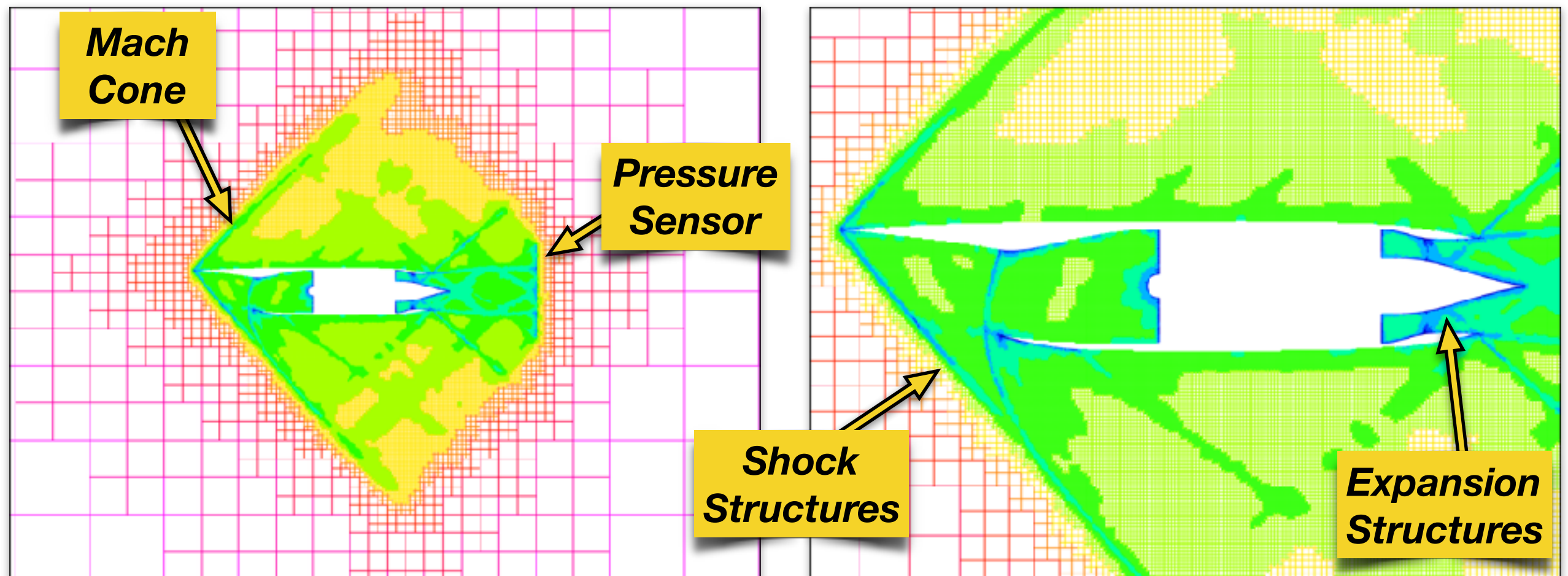- Mass flow rate quickly converges and continues to converge through each refined mesh
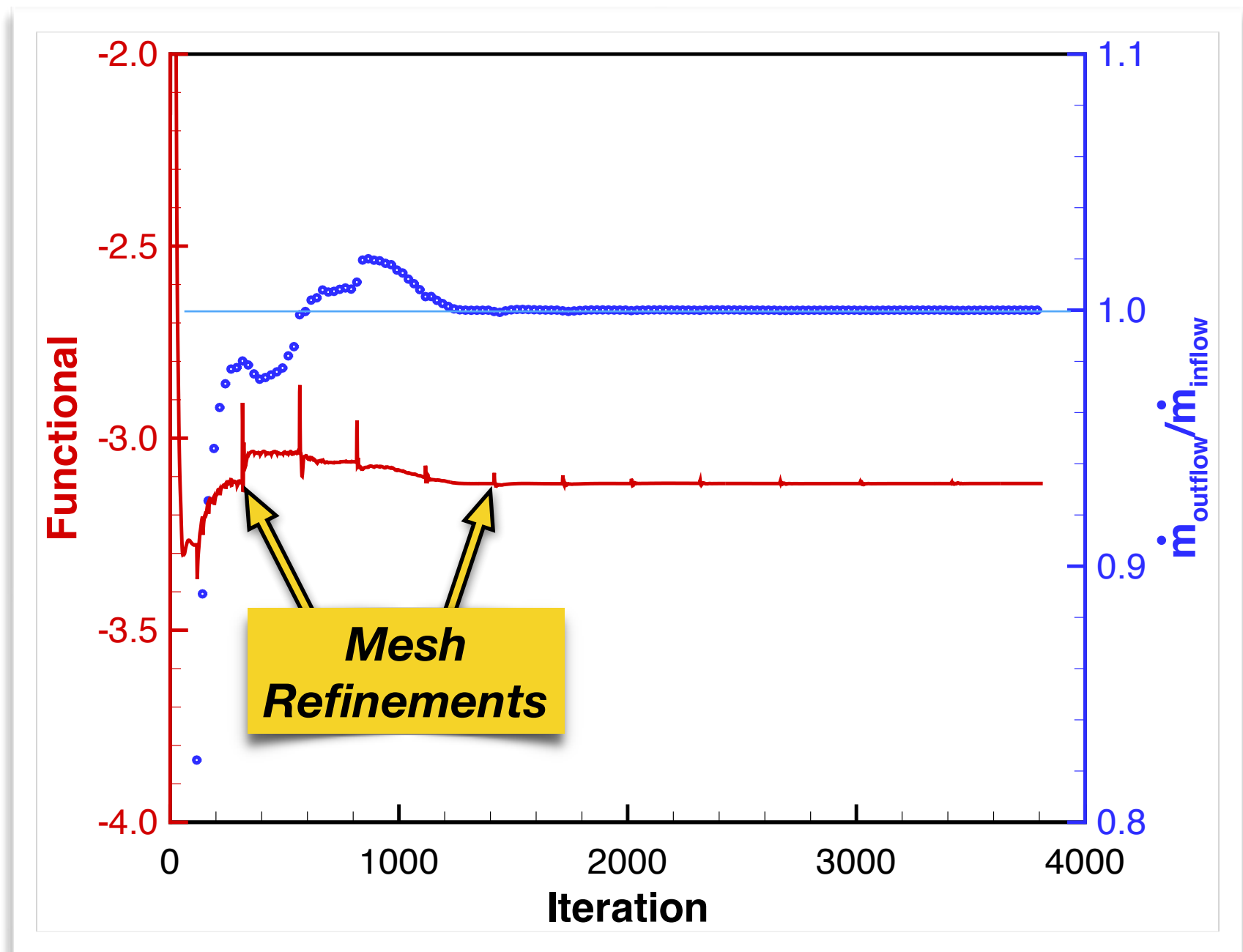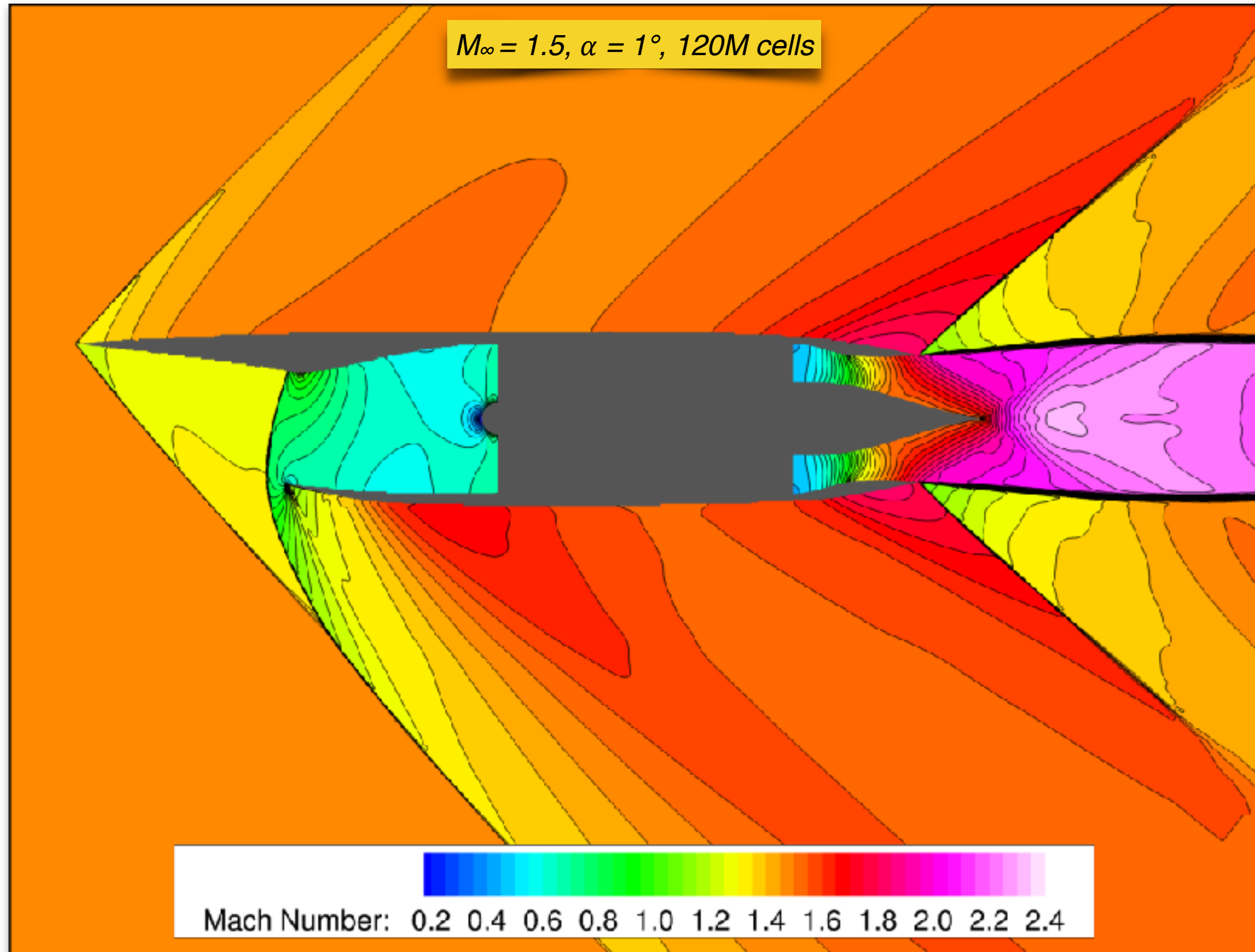

Mesh Refinements

Turbojet - Adaptively Refined Mesh

- Colors represent cells of same level of refinement

- Mesh was refined at surface, within Mach cone of influence, at shock and expansion structures, and at plume shear layer influencing pressure sensor

Turbojet - Matched Mass Flow Rates

$M_\infty = 1.5, \alpha = 1°, 120M \ cells$

Mach Number: 0.2 0.4 0.6 0.8 1.0 1.2 1.4 1.6 1.8 2.0 2.2 2.4

Scramjet in Hypersonic Flow

- Multiple ramp inlet and outlet, flow through burner remains supersonic

- Mach 5.0 freestream, 2° angle of attack

- Subsonic inflow / outflow boundary conditions not applicable

- Original full state with Riemann solver (`SurfBC`) boundary condition applied

- Mesh was refined at surface, within Mach cone of influence, shock and expansion structures, and plume shear layer influencing pressure sensor
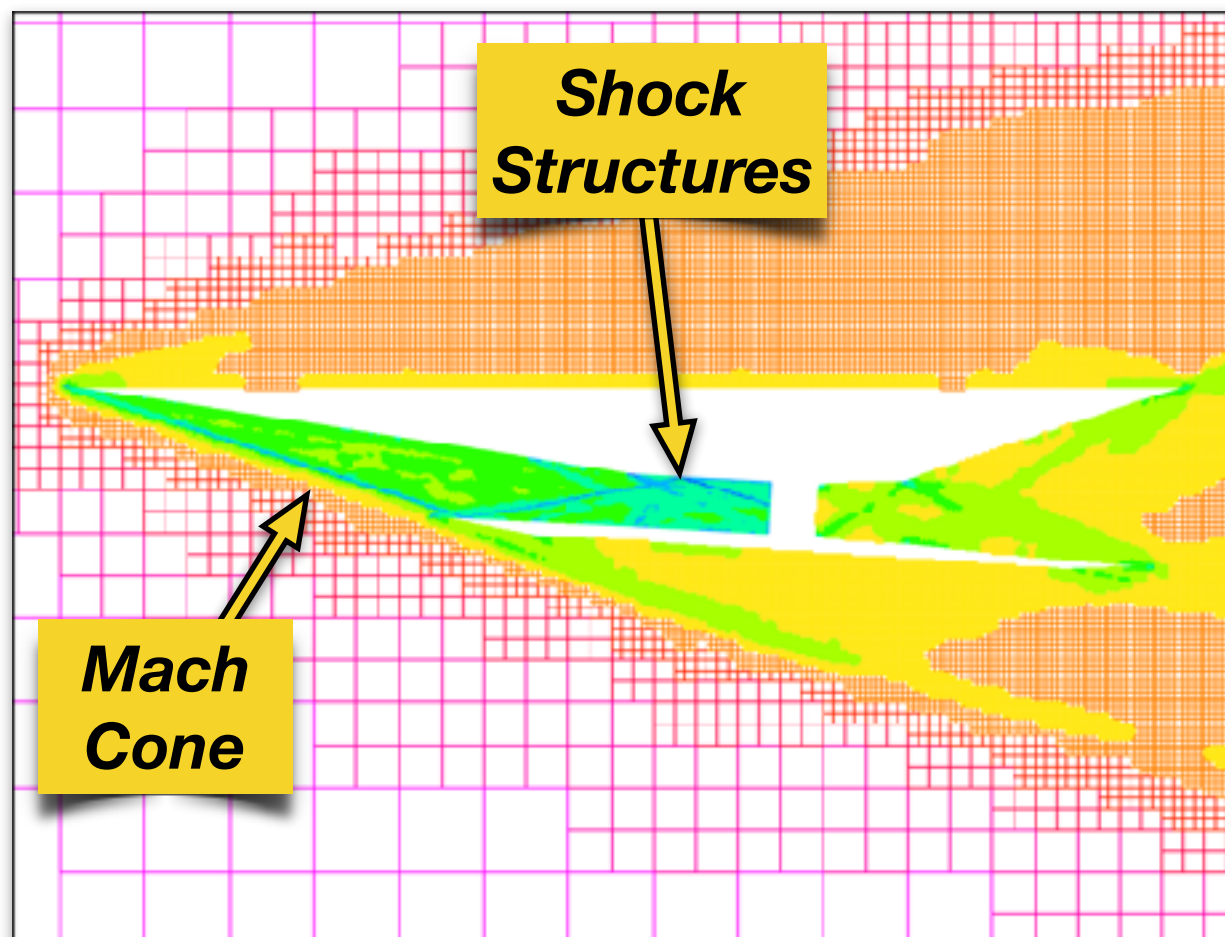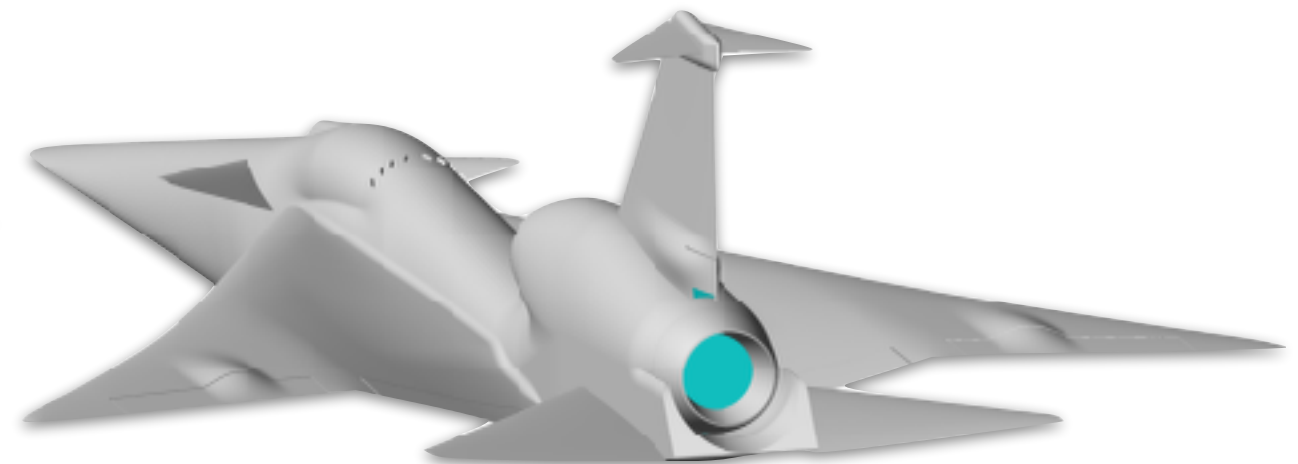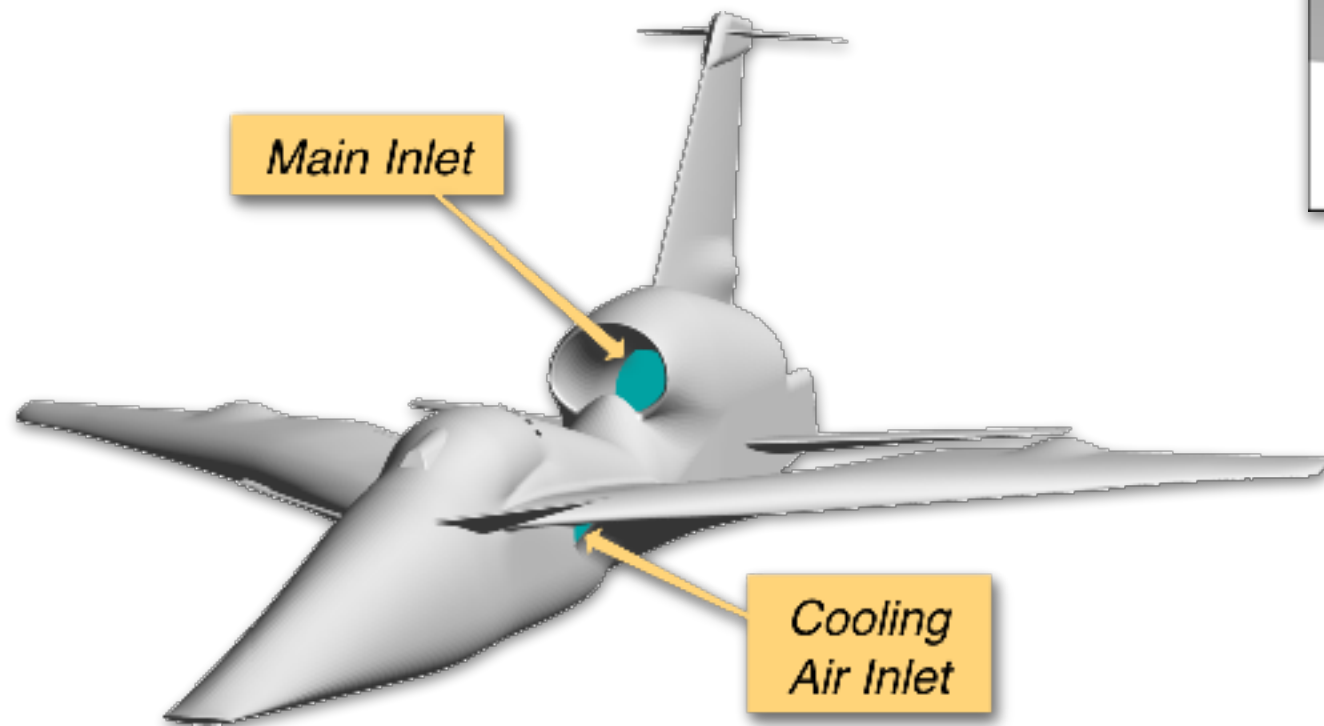

Shock Structures

Mach Cone

$M_\infty = 5.0, \alpha = 2°, 220M$ cells

Mach Number: 2.0 2.4 2.8 3.2 3.6 4.0 4.4 4.8 5.2
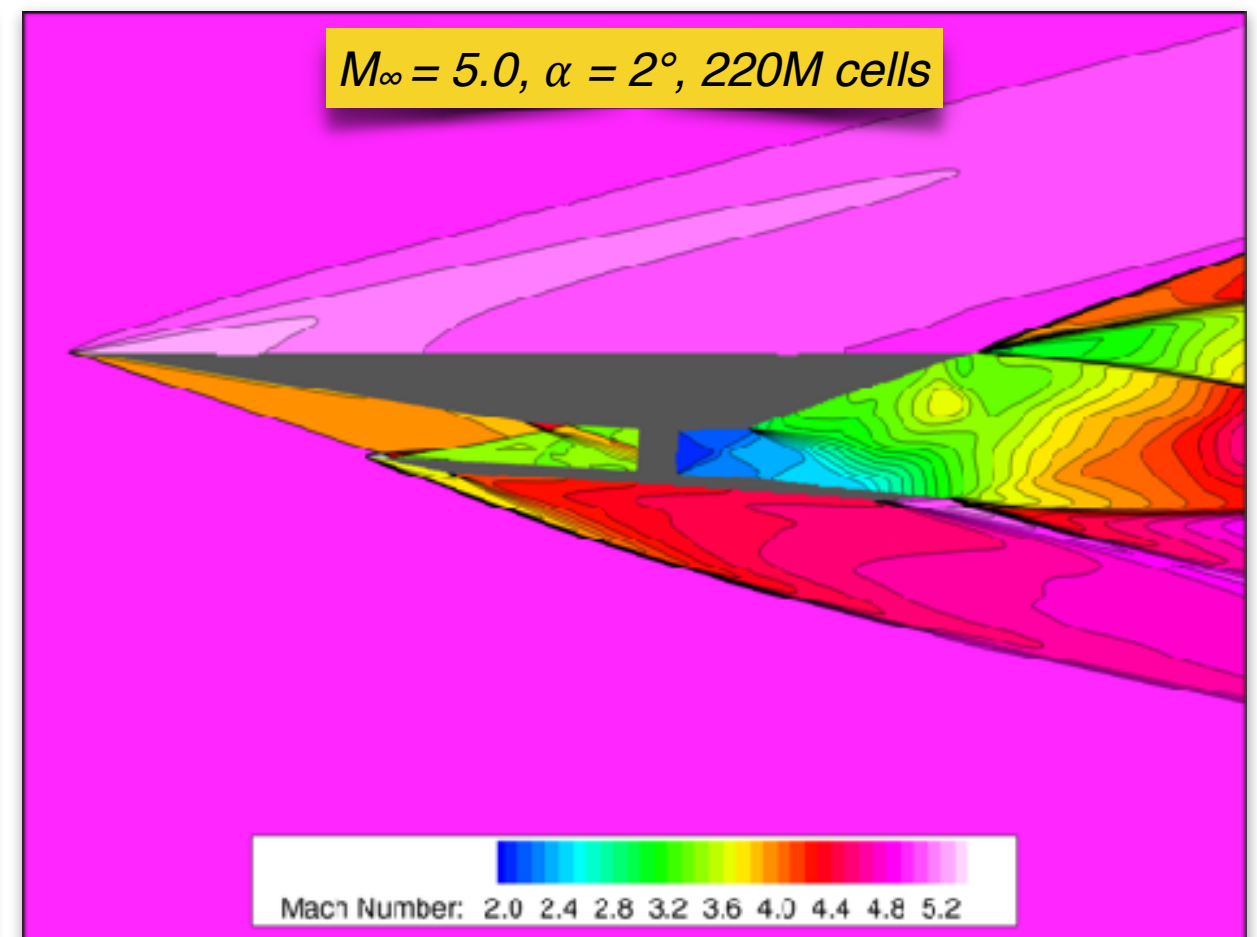
Low Boom Supersonic Demonstrator

- Realistically complex geometry

- Mach 1.4 freestream, 2.15° angle of attack

- 3 inlets and 3 exhausts

Tail Base Vent

Main Exhaust Nozzle

Under Nozzle Exhaust Port
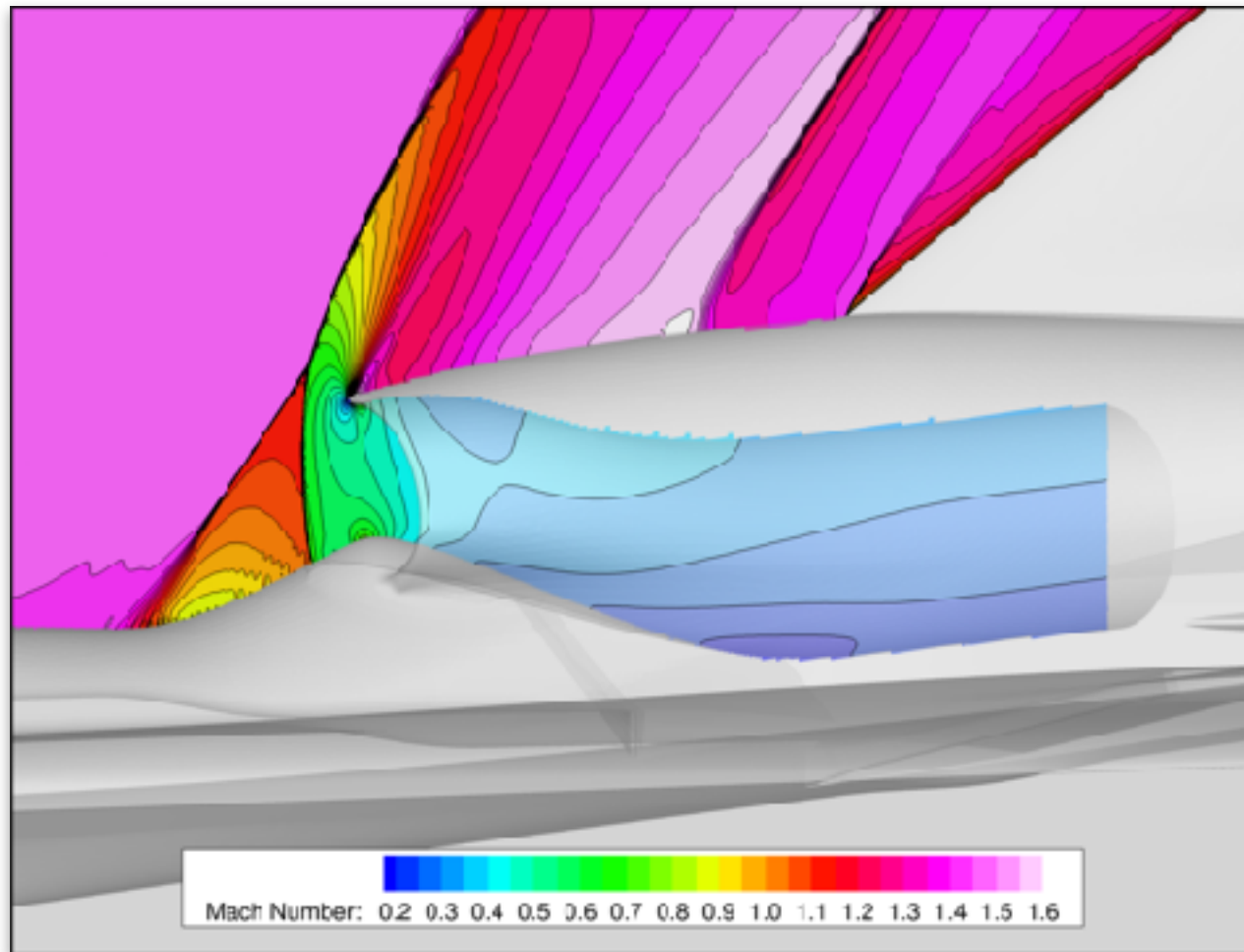
Main Inlet

Cooling Air Inlet
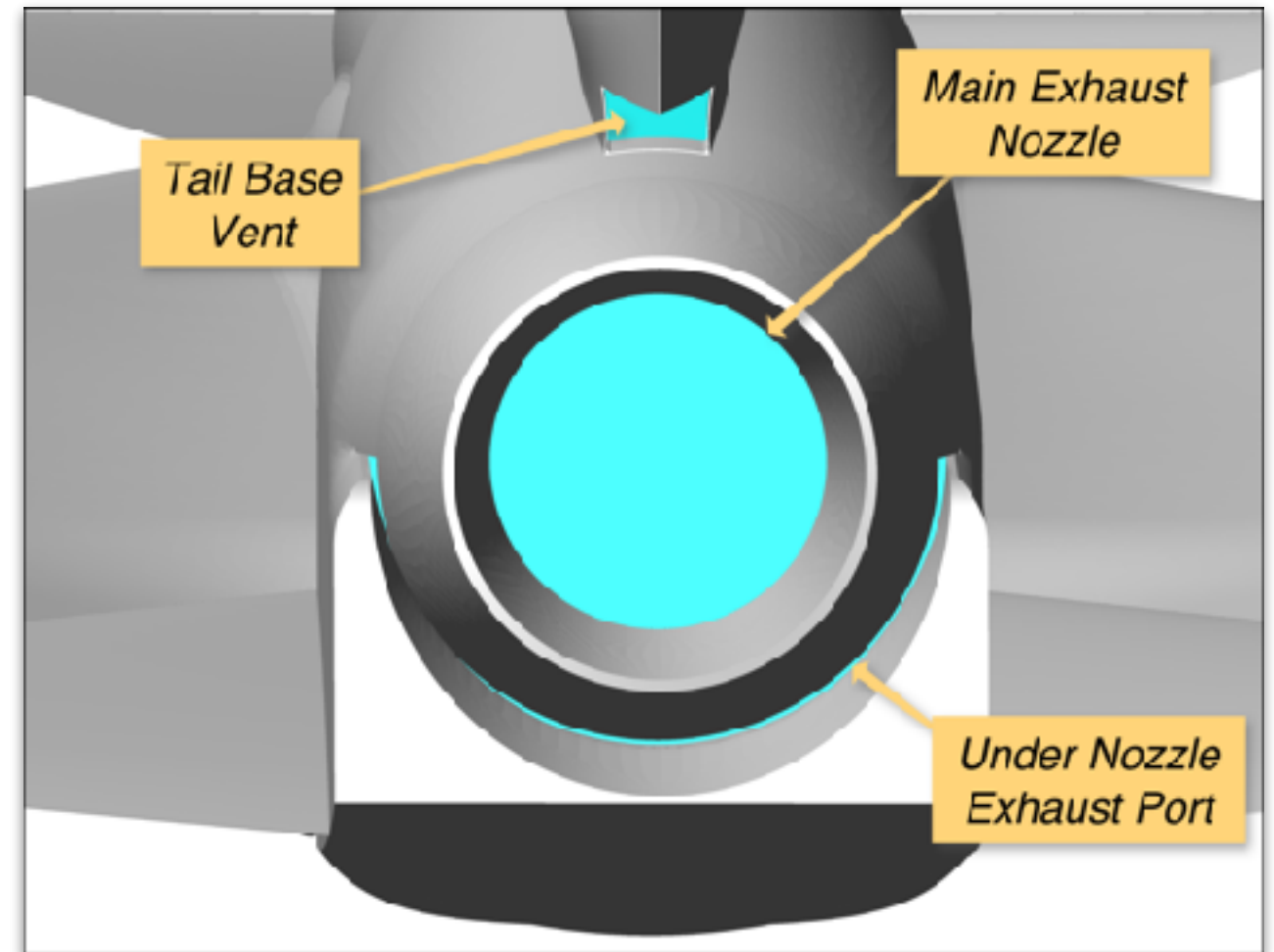
Low Boom Aircraft - Adaptively Refined Mesh

- Functional was aircraft drag

- Colors represent cells of same level of refinement

- Mesh was refined at surface, within Mach cone of influence, and at shock and expansion structures

Mach Cone
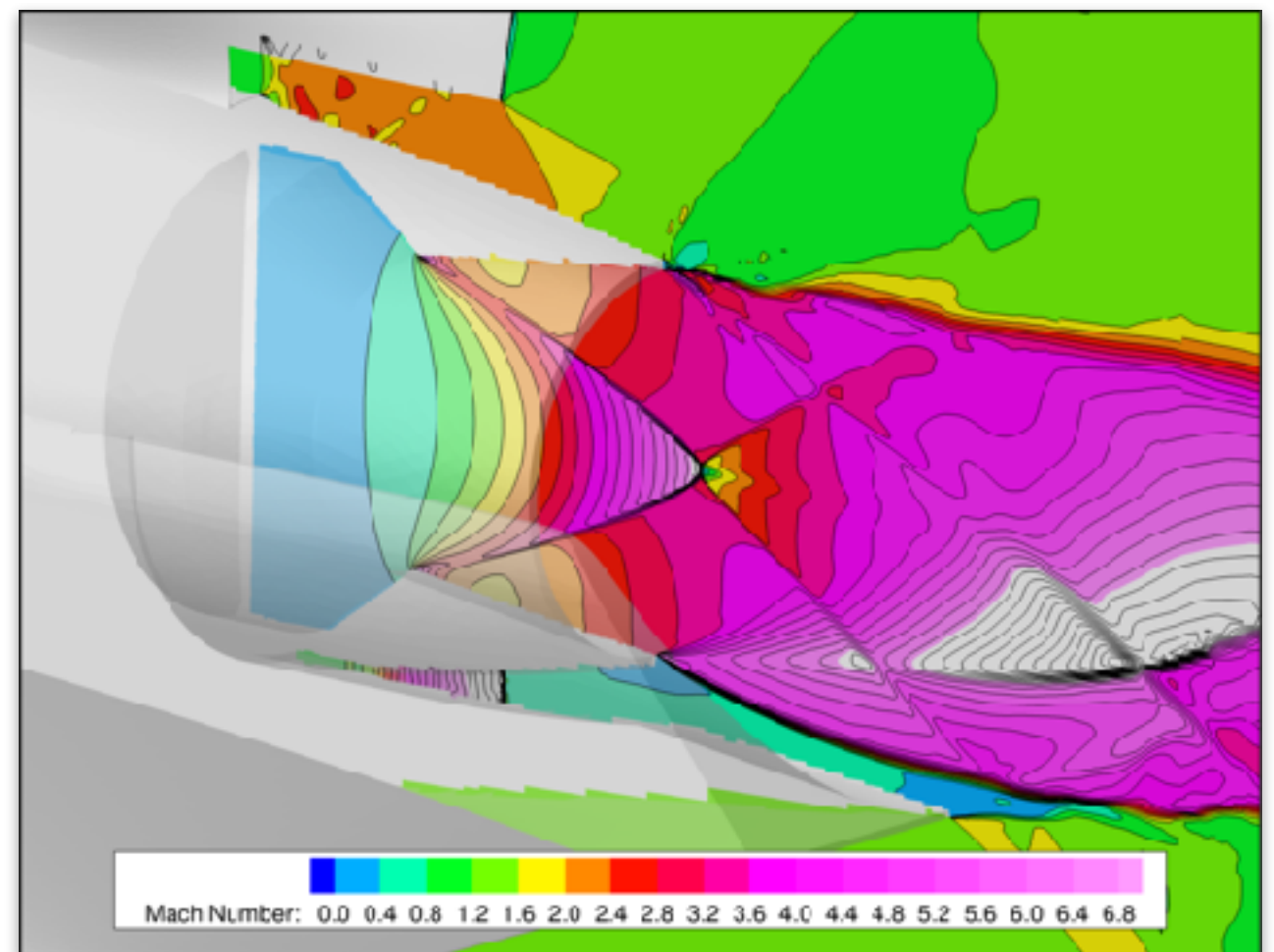
Shock Structures

Shock/Expansion Structures
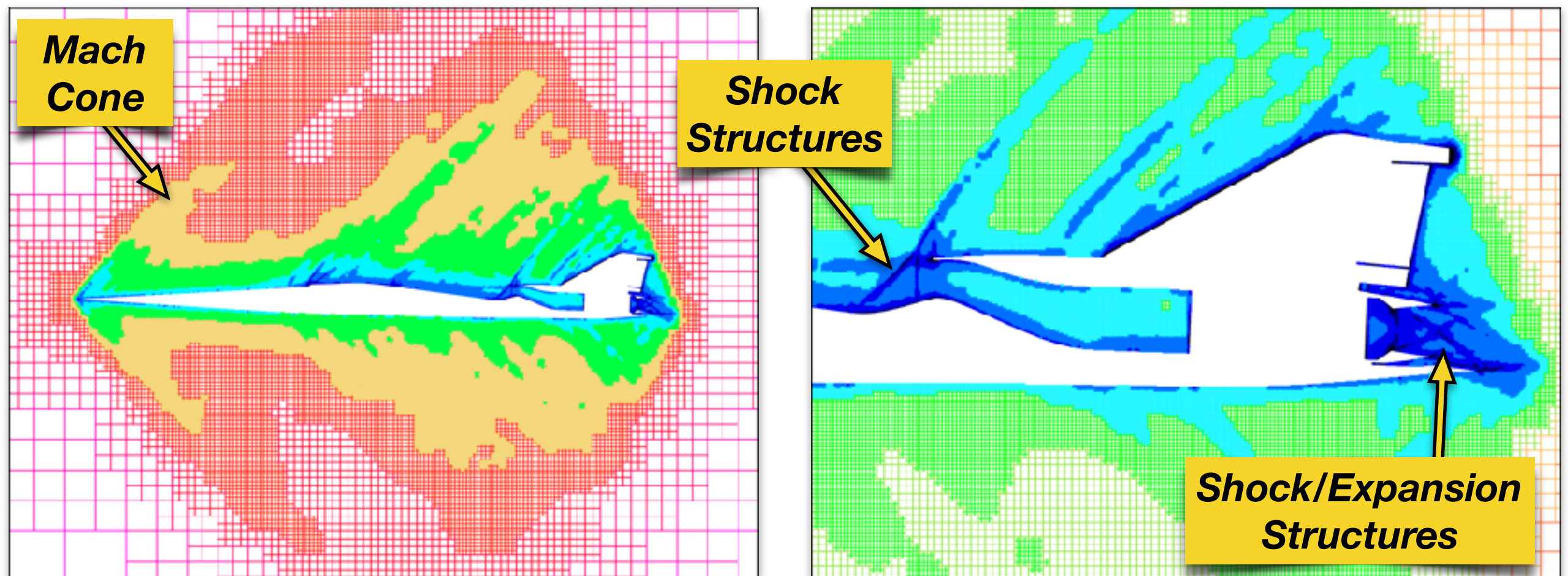
Low Boom Aircraft - Example Solution

$M_\infty = 1.4, \alpha = 2.15°,$
$70M$ cells (half-body mesh)

- Underwing inlet geometry is not fully realized

- Safeguards were active in these inlets (solid wall to not allow reverse flow)

Summary and Ongoing Work

- Four new subsonic inflow/outflow boundary conditions implemented to improve modeling of propulsion systems

- Robust mass flow rate control implemented for both inflow and outflow

- Demonstrated on notional propulsion systems in flight regimes ranging from subsonic to hypersonic

  - adjoint-driven mesh refinement demonstrated with all propulsion boundary conditions

  - new boundary conditions verified mesh convergence studies on notional examples

- Demonstrated on realistically complex low boom aircraft

- *Ongoing work*

  - *Implement additional functionals appropriate for propulsion systems*

  - *Extend design framework to include new propulsion boundary conditions and functionals*
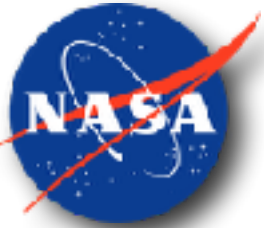
Acknowledgements

- NASA ARMD Commercial Supersonic Technology (CST) Project provided funding

- NASA Advanced Supercomputing (NAS) Center provided computing resources

- Other colleagues in Computational Aerosciences Branch